Cape Peninsula
University of Technology

# DECOMPOSITION COORDINATING METHOD FOR THE SOLUTION OF A MULTI-AREA POWER SYSTEM DYNAMIC OPTIMISATION PROBLEM INCORPORATING DISTRIBUTED GENERATION SOURCES

**by**

LITHA MBANGENI

**Thesis submitted in fulfilment of the requirements for the degree**

**Doctor of Engineering:** Electrical Engineering

**in the Faculty of** Engineering

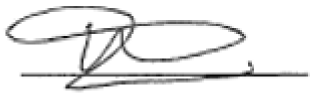
**at the Cape Peninsula University of Technology**

**Supervisor:** Prof. Senthil Krishnamurthy

**Bellville**
Feburay 2026

# DECLARATION

I, Litha Mbangeni, declare that the contents of this dissertation/thesis represent my own unaided work, and that the dissertation/thesis has not previously been submitted for academic examination towards any qualification. Furthermore, it represents my own opinions and not necessarily those of the Cape Peninsula University of Technology.



**15/02/2026**

**Signed** **Date**

## ABSTRACT

This research investigates the optimal coordination of wind and thermal power generation in single- and multi-area systems using advanced economic dispatch formulations and optimization methods. A comprehensive literature review was conducted to examine existing problem formulations, methodologies, and algorithms applied to hybrid power systems, including wind-thermal, wind-diesel, wind-PV, and hydrothermal configurations.

The study first develops a rigorous mathematical formulation for the wind-thermal economic emission dispatch (WEED) problem in a single-area power system, explicitly incorporating operational constraints and emission considerations. The formulation is then extended to a multi-area wind-thermal economic dispatch (WTED) problem, where inter-area power exchange and tie-line constraints are modelled to reflect realistic interconnected grid conditions.

Two optimization techniques are developed and implemented: the Lagrange multiplier method, representing a conventional analytical approach, and the Particle Swarm Optimization (PSO) method, representing a modern metaheuristic algorithm. Both methods are applied to the single-area and multi-area WTED problems. To facilitate computational analysis, MATLAB-based software is developed for each approach, including a decomposition-based framework for the multi-area case to enhance scalability and solution efficiency.

The developed methods are tested on standard IEEE benchmark systems, and the obtained results are compared with existing solutions reported in the literature. The comparative analysis demonstrates the accuracy, robustness, and computational efficiency of the proposed approaches. The findings provide valuable insights into optimizing hybrid wind-thermal power systems, advancing sustainable, economically efficient power generation strategies.

**Keywords**: Multi-region wind-conventional economic emission dispatch, decomposition approach, renewable energy, and optimization techniques.

# ACKNOWLEDGEMENTS

**I wish to thank:**

- My wife, Khayakazi Mbangeni, whose unwavering support made the completion of my studies possible. Without her love, strength, and encouragement, this achievement would not have been attainable.
- To my children, Khanyisile, Owesipho, Ithaqama, and Lwaluhle, thank you for your patience, understanding, and the many sacrifices you made during the period of my studies.
- My parents, Mandzolwandle Christopher, Fezeka Felicia Mbangeni, and my sister Zintle Tablsa Mbangeni, for their constant support, patience, endurance, and understanding throughout the journey of my studies.
- The Sipho Mbangeni Family, for always believing in me and affirming that one day, I would become Dr. Mbangeni.
- Professor Senthil Krishnamurthy, for his invaluable guidance, support, and encouragement in all areas. This thesis would not have been possible without his dedication and mentorship.

# DEDICATION

Late grandfather Sipho Mbangeni and Grandmother Julia Matshezi Mbangeni. I also dedicate this achievement to the late Professor Raynitchta Tzoneva. Without you, Professor, I would not be where I am today. You guided me from my Bachelor's degree, through my Master's, and now to the completion of my DEng. You were more than a supervisor to me; you were a mentor, a mother, and a constant source of strength and inspiration. I wish with all my heart that you were still alive to witness this milestone. Your belief in me shaped the person and scholar I have become. May your dear soul rest in eternal peace. Your legacy lives on in my journey.

# TABLE OF CONTENTS

LIST OF FIGURES

**LIST OF TABLES**

## GLOSSARY

## ABBREVIATIONS AND ACRONYMS

| Acronyms/Abbreviations | Explanation |
|---|---|
| ABC | Artificial Bee Colony |
| AI | Artificial Intelligence |
| ANN | Artificial Neural Network |
| ANN/GA/PL | Artificial Neural Network with Genetic Algorithm and Priority List |
| CC | Cluster of Computers |
| CCP | Chance Constrained Programming |
| CDF | Cumulative Distribution Function |
| CE-NSGA-II | Controlled Elitist-Non-Dominated Sorting Genetic Algorithm II |
| CMA-ES | Covariant Matrix Adaptation Evolution Strategy |
| DED | Dynamic Economic Dispatch |
| DER | Distributed Energy Resources |
| DSR | Down Spinning Reserve |
| ED | Economic Dispatch |
| EED | Economic Emission Dispatch |
| EP | Evolutionary Programming |
| EV | Electric Vehicle |
| FF | Fuel Function |
| FFG | Fossil Fuel Fired Generator |
| GA | Genetic Algorithm |
| GT | Game Theory |
| H-MCSA-DE | Hybrid Modified Cuckoo Search Algorithm and Differential Algorithm |
| HRES | Hybrid Renewable Energy System |
| HS | Harmony Search |
| IO | Iterative Optimisation |
| IOT | Iterative Optimisation Technique |
| LR | Lagrangian Relaxation |
| MADED | Multi-Area Dynamic Economic Dispatch |
| MAED | Multi Area Economic Dispatch |
| MAEED | Multi Area Economic Emission Dispatch |
| MAWTEED | Multi Area Wind Thermal Economic Emission Dispatch |
| MATLAB | Matrix Laboratory |
| MDT | Minimum Down Time |
| MILP | Mixed Integer Linear Programming |
| MIP | Mixed Integer Programming |
| ML | Machine Learning |
| MLT | Mean Learning Technique |
| MOEA | Multi-Objective Evolutionary Algorithm |
| MUT | Minimum Up time |
| MW | Megawatt |
| NSGA-II | Non-Dominated Sorting Genetic Algorithm II |
| OCD | Optimal Conditional Decomposition |
| PDF | Probability Distribution Function |
| PDIP | Primal Dual Interior Point |
| PL | Priority List |
| PSO | Particle Swarm Optimisation |
| PV | Photovoltaic |

| | |
|---|---|
| QNM | Quasi-Newton Method |
| RES | Renewable Energy Source |
| RTDS | Real Time Digital Simulator |
| RVCA | Real-Valued Cultural Algorithm |
| SA | Simulated Annealing |
| SED | Static Economic Dispatch |
| SMCS | Sequential Monte Carlo Simulation |
| SMODE | Summation-based Multi-Objective Differential Evolution |
| USR | Up Spinning Reserves |
| WECS | Wind Energy Convention System |
| WDF | Weibull Distribution System |
| WPDF | Weibull Probability Distribution Function |
| WTEED | Wind Thermal Economic Emission Dispatch |
| $P_w$ | Power delivered by a wind turbine [MW] |
| $P_R$ | Rated power output of the wind turbine[MW] |
| $v_{in}$ | Cut in wind speed[m/s] |
| $v_r$ | Rated wind speed[m/s] |
| $v_{out}$ | Cut out wind speed[m/s] |
| ρ | Air Density[kg/m$^3$] |
| A | Rotor swept area |
| $C_p$ | Power coefficient of a wind turbine |
| $F(v)$ | Wind probability distribution function |
| $k, c$ | Weibull distribution shape and scale parameters |
| $F_C$ | Total generation cost[$/h] |
| $C_i$ | Cost coefficient of a generator i |
| $P_{Gi}$ | Output power of thermal generator I [MW] |
| $P_{Di}$ | Power demand in area i [MW] |
| $P_L$ | Transmission line losses [MW] |
| $B_{ij}$ | Transmission loss coefficients |
| $E_i$ | Emission of a generator i[kg |
| $a_i, b_i, c_i$ | Quadratic cost coefficient for generator I |
| $\alpha_i, \beta_i, \gamma_i$ | Emission coefficients for $CO_2, NO_x, SO_2$ |
| **λ** | Lagrange multiplier |
| N | Number of areas |
| $R_u, R_d$ | Ramp up and ramp down limits[MW/min] |
| $h$ | Price penalty factor |

# CHAPTER ONE
## INTRODUCTION

### 1.1 Awareness of the research problem.

Economic dispatch is essential to the operation of power systems, and its primary objective is to allocate power to consumers at minimum cost, subject to system constraints. (Wood & Wollenberg, 2012). This is Static Economic Dispatch (SED) if the dispatching time interval is a single time period, and Dynamic Economic Dispatch (DED) if the dispatching time interval spans multiple, continuous time periods. However, more research has shown that the use of ED in power can be more accurate and practical for monitoring system behaviour. (Cheng & Zhang, 2014).

Prior to the discovery of various alternative energy sources, the economic dispatch problem involved only conventional thermal energy generation. Due to a sudden rise in traditional energy sources, such as coal and oil, and atmospheric gases, several nations have supported research on renewable energy applications to diversify energy sources. Wind is one of the alternative energy sources among these projects. One MAWTEED study of interest is MADED (Multi-Area Dynamic Economic Dispatch). A publication (Soroudi & Rabiee, 2013) associated with this research is MADED: Multi-Area Dynamic Economic Dispatch. MADED considers wind power uncertainties, a power pool market, and hydrothermal generating units in each area, and utilizes Optimal Condition Decomposition (OCD) with parallel computing. It is implemented for two interrelated power resources. The successful implementation of a multi-area power system ensures continuous power availability. However, the frequency fluctuates slightly due to greater interconnected power system deviations and variations in power transmission to other sub-areas. This can be achieved by interconnecting tie-lines between the sub-areas, as shown in Figure 1.1. Also, the use of thermal units that can be used as reserves (Up-Spinning reserves (USR) and Down Spinning Reserve (DSR) to compensate for the wind uncertainties.

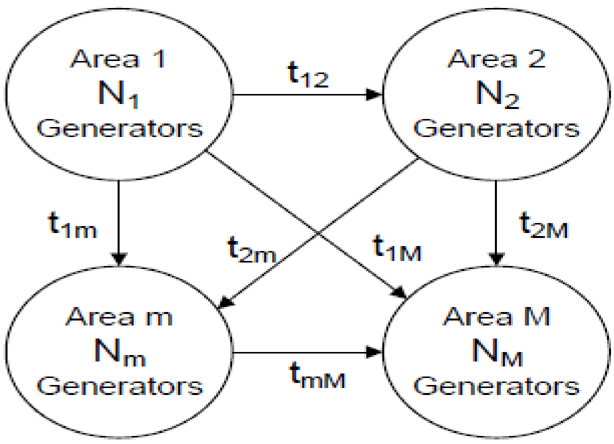



**Figure 1.1:** **Model of a multi-area power system with tie-line power transfer**

(Krishnamurthy & Tzoneva, 2016)

Where

| | |
|---|---|
| $N_m$ | The number of generators belonging to the area $m$ |
| $M$ | A number of interconnected areas |
| $t_{1\mathrm{m}}$ | Number of tie-lines from area 1 to area $M$ |
| $t_M$ | Number of tie-lines connecting areas |

Environmental problems and fuel consumption can be reduced significantly with renewable energy sources. (Liang et al., 2015). However, the uncertainties in wind speed and its uncontrollability create uncertainty in wind power production. Wind power is fundamentally stochastic compared to other conventional energy sources, meaning its use must be evaluated at a larger, more aggregate scale within complex systems.

In this regard, (Xia & Elaiw, 2010) Also offers a survey of mathematical optimization solutions that have addressed the problem involving distributed generation resources with effective results, such as the lambda iterative method, Lagrange relaxation, as well as Artificial Intelligence methods, which consist of artificial neural networks, Genetic Algorithm (GA), Simulated Annealing (SA), Evolutionary Programming (EP), Particle Swarm Optimization(PSO).

(Jeyakumar et al., 2006), done a comparison study of PSO with other optimization methods in different types of Economic dispatch (ED). It was observed that PSO convergence was better than that of Classical Evolutionary Programming (CEP) for MAED. It is simple and easy to converge to the global optimum. This algorithm can be used to determine the optimal power allocation of generating units in each area.

The MAWTEED problem has many variables, nonlinear equations, and high dimensions, making calculations very complex. The MAWTEED problem can be very complex; hence, it is necessary to formulate the problem and develop an algorithm and software to solve it using the Lagrange decomposition method.

The MAWTEED is solvable via the Lagrange decomposition coordinating approach, which is one of the optimal methods for parallelizing a solution. It takes a larger, overall problem and creates sub-problems whose solutions, when added together, equal the overall solution; thus, optimality is preserved.

Based on the above, this thesis investigated a solution to the problem for MAWTEED using the decomposition method

## 1.2 Statement of the research problem

Replacing thermal generating units with renewable energy sources (RES) reduces atmospheric gas emissions and fuel consumption(Wang & Singh, 2007). However, wind plants provide much more uncertain capacity than thermal generating units and are much more nonlinear, making predictions for generation output less certain. Therefore, an electric power system largely depends on thermal generation for the time being. The integration of wind turbines with existing thermal generating units can pose challenges for power system balancing. If wind power is available, it can reduce the need for conventional generating capacity, freeing it for reserve purposes. On the other hand, if there is a sudden reduction in wind power, it directly affects the system frequency between generation and load. Wind energy can significantly shorten the lifespan of thermal generators, while these units play a vital role as reserves to compensate for wind uncertainties. Therefore, wind and thermal generation coordination constitutes a more complex optimization problem for creating a cost-effective dispatch schedule.

Power systems are interconnected because interconnected areas are more reliable and can be operated at a lower cost than isolated areas. A change in load demand in one area can be accommodated by interchanging power between the regions via tie lines. Suppose a generating unit is lost in one of the interconnected areas. In that case, the connected regions increase their power outputs to compensate for the lost plant until the reserve or standby generator is brought into service.

It is necessary to address the difficulties above using methods applicable to the conditions of deregulated power system operation and to reduce the complexity and data communication between interconnected systems.

The research problem can be divided into the following sub-problems.

- Design-based sub-problems
- Implementation sub-problems.

## 1.3 Design-based problems

1.3.1 Sub-problem 1: Literature review of the different problems relative to the single area and the MAWTEED problem.

1.3.2 Sub-problem 2: Mathematical formulation of the single area and MAWTEED problem relative to the quadratic fuel cost of thermal generators and the Weibull probability distribution function of wind generation.

1.3.3 Sub-problem 3: Mathematical formulation of the single area WTEED problem relative to Lagrange's method.

1.3.4 Sub-problem 4: Mathematical formulation of the PSO method relative to the single area WTEED Problem.

1.3.5 Sub-problem 5: Mathematical formulation of the decomposed-coordinating Lagrange method relative to how to solve the MAWTEED Problem.

1.3.6 Sub-problem 6: Mathematical formulation of PSO relative to how to solve the MAWTEED Problem.

## 1.4 Sub-problems of implementation

1.4.1 Sub-problem 1: Software Implementation relative to Lagrange's methods to solve the WTEED Problem based on the quadratic fuel cost of thermal generators and the Weibull Probability Distribution function of wind power generators.

1.4.2 Sub-problem 2: Software Implementation relative to the PSO method for the single area WTEED Problem based on quadratic fuel cost of thermal generators and Weibull probability distribution function of wind plants.

1.4.3 Sub-problem 3: Software Implementation relative to Lagrange's decomposition-coordinating method to solve the MAWTEED Problem based on the quadratic fuel cost of thermal generators and the Weibull probability distribution function of wind turbines

1.4.4 Sub-problem 4: Software Implementation relative to the PSO algorithm to solve the MAWTEED Problem.

## 1.5 Research aim and objectives

### 1.5.1 Research Aim

- The research aim focuses on developing the methods and algorithms for the wind-thermal economic emission dispatch problem solution for single and multi-area power systems. Develop a decomposition-coordinating method for the solution of the multi-area dispatch problem using Lagrange's algorithm. The software is implemented sequentially on one computer.

### 1.5.2 Objectives

- To survey the literature of already existing problem formulation, techniques, and algorithms of wind-thermal, wind-diesel, wind-PV, and hydrothermal economic dispatch problems.
- To mathematically formulate the wind-thermal economic emission dispatch problem with constraints for single-area and multi-area power systems.
- To develop the Lagrange's method for the wind-thermal economic dispatch problem for both single area and MAWTEED.
- To create a Particle Swarm Optimisation (PSO) method for both single area and MAWTEED.
- To develop MATLAB software for the developed Lagrange's and PSO methods for the wind-thermal economic dispatch problem for the single area power system.

- To develop MATLAB software for the Lagrange's and PSO methods for the wind-thermal economic dispatch problem for multi-area power systems through decomposition.
- To test the developed software for the standard IEEE benchmark models and compare the results with the ones already existing in the literature.

### 1.6 Hypotheses

The hypotheses related to the applicability of methods and algorithms to solve the single-area and multi-area wind-thermal dynamic economic dispatch problem have been investigated and can be stated as:

- The Weibull probability density function represents the best fit for the wind power plant formation. The quadratic fuel cost curve is assumed for the conventional generators (Coal, gas, and hydro plants).
- The methods and algorithms of the economic dispatch problem are based upon classical, heuristic, and hybrid optimization techniques, which ascertain that the hybrid algorithm possesses computational efficiency, providing better solutions and ease of implementation in practical applications.
- The recent findings of the multi-area dispatch problem postulate that the decomposed complex nature of dispatch problems for a vast interconnected area does, in fact, provide a solution concerning tie-line powers of the interconnected system.
- The Lagrange method has been developed for the economic emission dispatch problem for single-area dispatch purposes, and subsequently, the dispatch problem for multi-area dispatch has been created.
- The PSO method has also been developed and tested for both single-area and multi-area dispatch applications.
- Finally, results comparison can be made between the two developed methods (Lagrange's PSO).

### 1.7 Delimitation of the research

The thesis focuses on classical and heuristic methods for solving the single- and multi-area emission economic dispatch problem. The boundary conditions of the MAWTEED problem considered in this thesis are as follows:

- Wind-PV, Diesel-thermal, and hydrothermal economic dispatch problems are not considered in this research work. The wind-thermal economic emission dispatch problem is formulated and implemented using classical and heuristic optimization methods.

- The hybrid optimization method is not considered in this research work. The classical (Lagrangian) and heuristic (PSO) methods are used to solve the single- and MAWTEED problems.
- The cubic cost function model of the conventional economic dispatch problem cannot be considered. The quadratic cost function is used to formulate the traditional economic dispatch problem.
- The decomposition-coordinating method is used to solve the complex multi-area economic emission dispatch problem. For the single-area economic emission dispatch problem, the decomposition-coordination method is not applicable without tie-line powers.

## 1.8 Motivation of the research

- Most of the electricity is currently generated by thermal plants. These plants affect communities in different ways. Firstly, the gases that are discharged, i.e.,$CO_2$, and $NO_x$ are affecting global warming. Secondly, the raw material used is expensive.
- Other generating units like hydro power are better than fossil-fuel, because they use rainwater. However, they have drawbacks, such as the location of these plants being far from load centres and the need for long transmission lines to transport power over long distances, which adds to the cost of building new transmission lines for interconnected systems.
- The integration of renewable energies into the existing thermal plants can reduce the difficulties mentioned above. These renewables include wind, solar, and other sources, and they use natural gas to generate electricity. The problem with these energies is that they are less controllable than other generating units. The interconnection of thermal and wind renewable energy reduces the cost of production, maximizes reliability, and provides better operating conditions, such as reserve sharing and improved stability.

- Given the literature review, the following is true. Few decomposition-coordination approaches exist to solve the MAWTEED problem; therefore, the new area of deregulated economic dispatch for interconnected systems warrants exploration. In addition, other techniques and existing software enable the MAWTEED problem to be solved on a single computer. Therefore, this project constructs the MAWTEED problem and solves it with a Lagrange decomposition-coordination approach.

## 1.9 Assumption

The assumptions are made as follows:

- Each area's power demand is equal to the summation of wind power and thermal power.
- The network losses must be considered; however, in terms of distance, the greater the distance, the more power lost.
- The conventional generators have a minimum and maximum output power, i.e., no power plant will be utilized due to fluctuations in wind generation output.
- A quadratic cost function exists as a convex second-order function.
- The system operator owns the wind generators; thus, the expected cost of wind-generated output does not need to be factored into the cost function.
- Uncontrollability of wind power needs to account for over- and under-generation of wind power estimates in the cost function (Hetzer et al., 2008)
- Tie line constraints should be included in the MAWTEED problem.
- The number of areas in the interconnected power system is based on the IEEE benchmark models found in the literature.
- The MAWTEED problem is solved sequentially.
- The number of tie-lines must be calculated using (Krishnamurthy & Tzoneva, 2016).

Where

$M$ =total number of areas

$N_T$ =Number of tie-lines

- Smaller stop size provides the global solution for the search problem using either classical or heuristic optimization methods

## 1.10 The deliverables of the thesis

The primary deliverables of the study can be categorized as follows:

- An assessment of the literature on the single-area and multi-area economic emission dispatch problem comparatively.
- Mathematical formulation of the economic dispatch problem
- Formulation of the single area Wind-Thermal Economic Emission Dispatch (WTEED) problem with a quadratic fuel cost function and Weibull Distribution Function (WDF) based wind power.
- Formulation of the MAWTEED problem with a quadratic fuel cost function and WDF-based wind power.
- Optimization methods developed to solve the economic emission dispatch problems
- Development of Lagrange's methods for the solution of the single area WTEED problems.
- Created optimization methods to solve the economic emission dispatch problems

- Developed Lagrange's methods to solve the single area WTEED problems.
- Developed the PSO method to solve the single-area WTEED problem.
- Developed Lagrange's decomposition-coordinating method to solve the MAWTEED problem.
- Developed the PSO method to solve the MAWTEED problem.
- Created software to solve the economic emission dispatch problems in the single-area and multi-area cases
- Software was created based on Lagrange's methods and subsequent algorithms to solve the single-area WTEED problems.
- Software was created based on the PSO method and subsequent algorithms to solve the single-area WTEED problems in a sequential approach.
- Software was created based on Lagrange's decomposition-coordinating method to solve the MAWTEED problem.
- Software was created based on the PSO method to solve the MAWTEED problem.
- Software was created for the standard IEEE benchmark problems, and results were compared to those found in other literature.

For the summary of these contributions to the study, see Chapter 7.

## 1.11 Chapter breakdown

This study comprises seven chapters, and they are structured as follows:

Chapter 1 highlights the problem statement and the study's background. The study's aim and objectives, problem description, hypothesis, research delimitation, and assumptions are included in the thesis statement.

Chapter two documents the literature about the single-area and multi-area economic emission dispatch problems. It documents what is known about the economic emission dispatch problem and the pros and cons of different heuristic, hybrid, classical, and metaheuristic algorithms for solving it.

Chapter three formulates the Wind-Thermal Economic Emission Dispatch (WTEED) single-area problem with a quadratic fuel cost function and a Wind Distribution Function (WDF) for wind. Furthermore, the methodology for the single-area economic emission dispatch problem using Lagrange's method is established. Three different IEEE test benchmark models were simulated in MATLAB for the algorithms.

Chapter four formulates and explains the PSO algorithm for the WTEED single-area problem. Three different IEEE test benchmark models were simulated in MATLAB for the algorithms.

Chapter five formulates and establishes Lagrange's method for the Wind-Thermal Economic Emission Dispatch problem in the multi-area context with a quadratic fuel cost function and WDF for wind. Three different IEEE test benchmark models were simulated in MATLAB for the algorithms.

Chapter six formulates and establishes PSO for the Wind-Thermal Economic Emission Dispatch problem in the multi-area context with a quadratic fuel cost function and WDF for wind. Three different IEEE test benchmark models were simulated in MATLAB for the algorithms.

Chapter 7 discusses the study's conclusions and deliverables. The results of the study, future work, and a listing of the author's published papers comprise the conclusion.

## 1.12 Conclusion

This chapter presents the research need, research aim, objectives, problem statement, background, rationale, assumptions, and hypothesis. The reader will be given a brief overview of the study's findings here, with a more detailed discussion reserved for Chapter Seven. Thus, this chapter reflects on the relevance of the study, based on findings from single- and multi-area wind-thermal economic emission dispatch problems. Furthermore, it supports the thesis's Literature Review in Chapter Two, demonstrating that the comparison of different approaches/algorithms to solve single- and multi-area problems is justified.

# CHAPTER TWO
# LITERATURE REVIEW

## 2.1 Introduction

Electric power is essential to human survival in day-to-day life and serves as a driving force for many economic sectors. Conventional generation has been the sole method of provision of electric power for decades. Conventional generation exposes many cost and emission disadvantages.

This chapter introduces various optimisation strategies and algorithms for the economic emission dispatch problem, considering wind-thermal systems in a multi-area context. The optimisation methods are either search or gradient-based, as shown in Figure 2.1. The various formulations of the economic dispatch problem in the literature can be grouped into single-area and multiple-area dispatch, as illustrated in Figure 2.2.

The aim of this chapter is to investigate the solutions to single- and multi-area economic dispatch problems in the literature. This is organised in such a way that there are three types of solutions to the economic dispatch problem: 1) the single-area wind-thermal problem; 2) the single-area hybrid renewable energy resources problem; 3) the multi-area economic dispatch problem.

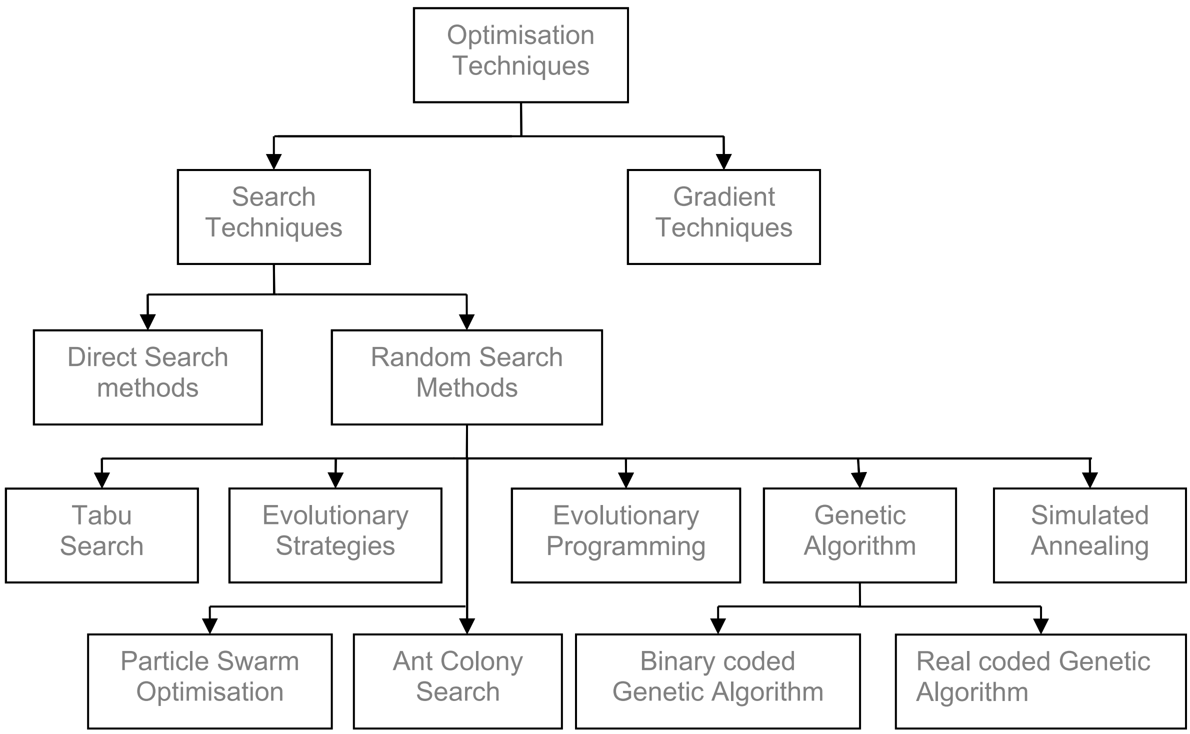



**Figure 2.1:** **Organisation of optimisation techniques (Kothari & Dhillon, 2012)**

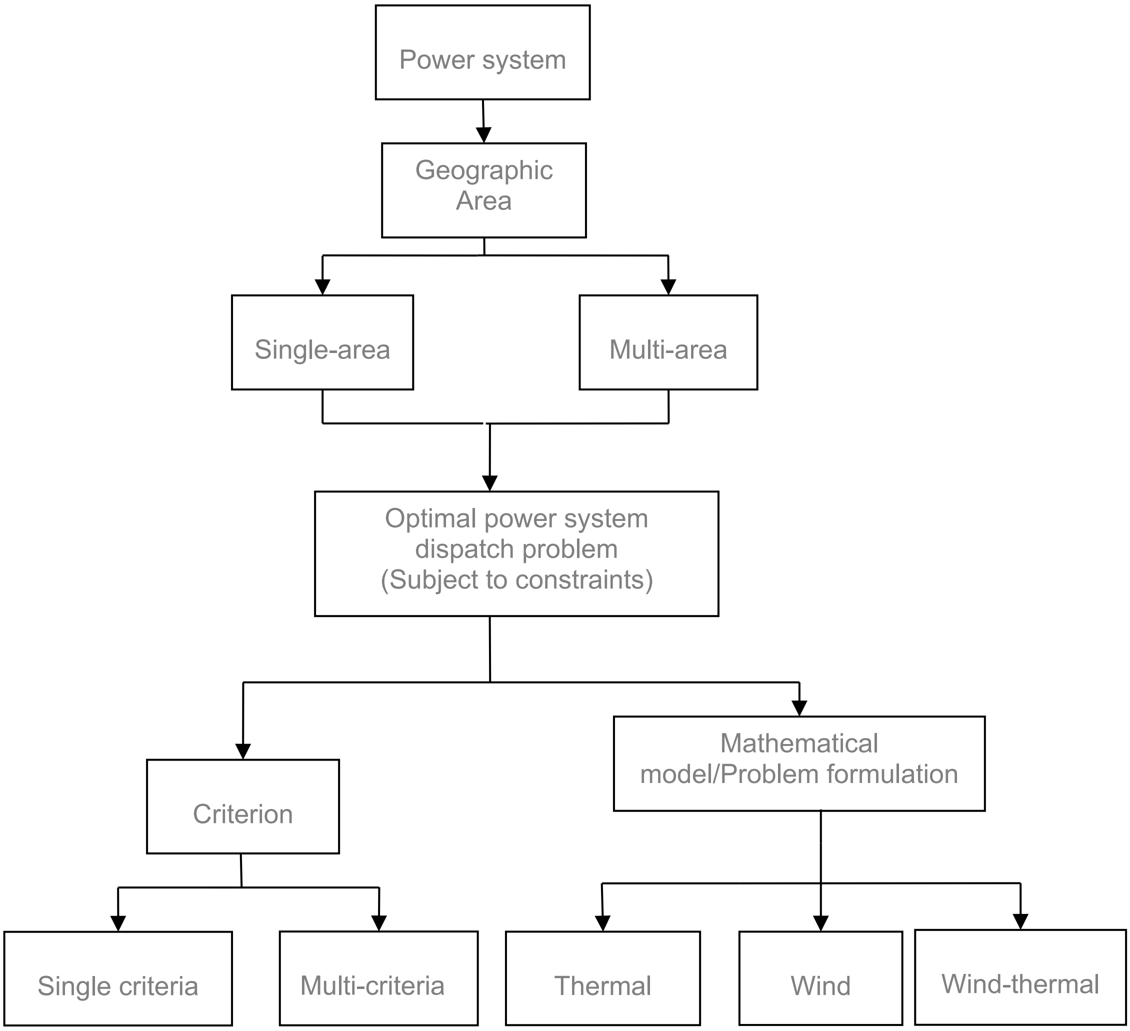



**Figure 2.2:** **Single-area and Multi-area power systems dispatch problem organisation**

## 2.2 Wind-thermal economic dispatch

The goal of the wind-thermal economic dispatch is to share power among conventional and renewable generating units and to minimize total operating cost, subject to equality or inequality design constraints. The time-varying wind-thermal economic dispatch is equivalent to a single-area and multi-area economic dispatch. The non-linear objective function can be a single or a two-objective function. One of the fundamental barriers to incorporating wind power into conventional economic dispatch is the uncontrollable, unpredictable nature of wind speed. Wind speed variability can best be modeled by a two-parameter Weibull probability density function, the most effective and reliable means(Jiang et al., 2015).

### 2.2.1 Problem formulation for single area wind-thermal economic dispatch

#### 2.2.1.1 Problem formulation for single-criterion economic dispatch.

In this part of the chapter, the problem statement of the wind-thermal economic dispatch problem with a single criterion function is given. The cost function for dynamic economic dispatch (DED) determines the minimum total operating cost of the entire generation system, subject to operational constraints. The total cost that the conventional and wind generators can represent is given by(Ghasemi et al., 2016).

$$F(P_g, P_w) = \sum_{i=1}^{N_G} F_i(P_{gi}) + \sum_{j=1}^{N_F}\sum_{k=1}^{N_W} W_{j,k}(P_{w_{j,k}}) \quad [\$/h] \qquad (2.1)$$

Where

| | |
|---|---|
| $F(P_g, P_w)$ | Total cost of all generating units |
| $N_G$ | Number of thermal generators |
| $N_F$ | Wind farm |
| $N_w$ | Number of wind generators |

The cost function of the thermal generator units is derived from a quadratic equation with valve-point loading effect, which is noted in (2.2) (Kothari & Dhillon, 2012). The thermal cost function can be with or without valve-point loading, as illustrated in Figure 2.3.

$$F_i(P_{gi}) = a_i + b_i P_{gi} + c_i P_{gi}^2 + \left| e_i \sin\left(f_i (P_{gi}^{\min} - P_{gi})\right)\right| \quad [\$/h] \qquad (2.2)$$

where $a_i, b_i,$ and $c_i$ is the fuel cost coefficient of the $i^{th}$ generator and $e_i$ and $f_i$ are valve point coefficients. The last term $W_{j,k}(P_{w_{j,k}})$Equation (2.1) represents the output power of the wind-generating units. This term can be defined using equation (2.3) (Hetzer et al., 2008).

$$W_{j,k}(P_{w_{j,k}}) = \varphi_{D_{j,k}} P_{E_{j,k}} + \varphi_{OE_{j,k}} \times (P_{A_{j,k}} - P_{E_{j,k}}) + \varphi_{UE_{j,k}} \times (P_{E_{j,k}} - P_{A_{j,k}}) \quad [\$/h] \qquad (2.3)$$

where

| | | |
|---|---|---|
| $P_{E_{j,k}}$ | Expected wind power generation of the turbine $j$ in a wind farm $k$ | [MW] |
| $P_{A_{j,k}}$ | Actual/available wind power generation of the turbine $j$ in a wind farm $k$ | [MW] |
| $\varphi_{D_{j,k}}$ | Direct cost coefficient of the wind turbine $j$ in a wind farm $k$ | [$/MWh] |
| $\varphi_{OE_{j,k}}$ | Overestimation cost coefficient of the turbine $j$ in a wind farm $k$ | [$/MWh] |
| $\varphi_{UE_{j,k}}$ | Underestimation cost coefficient of the turbine $j$ in a wind farm $k$ | [$/MWh] |

The thermal cost function is established with or without valve point loading (see Figure 2.3). Most economic dispatch studies address the combined economic emission dispatch problem from a linear-quadratic perspective, an idealization, since the incremental cost curves of generation units are assumed to be piecewise linear and continuously increasing. However, the actual input-output curves of thermal generating units are highly nonlinear, with fuel cost curves exhibiting valve-point loading. For instance, (Gernald & Aaron, 1958) demonstrate that valves permit economic adjustment in parallel turbines' loadings with one turbine operable under governor control for slight load demand adjustments so that power generation meets up-to-par standards with no incremental cost adjustments needed. But in actuality, a multitude of steam valves exist within generating stations. Thus, the thermal cost function, including the valve-point effect, is a truly nonlinear function that captures the true nature of the dynamic wind-thermal economic dispatch problem and thus operates within the cost function.

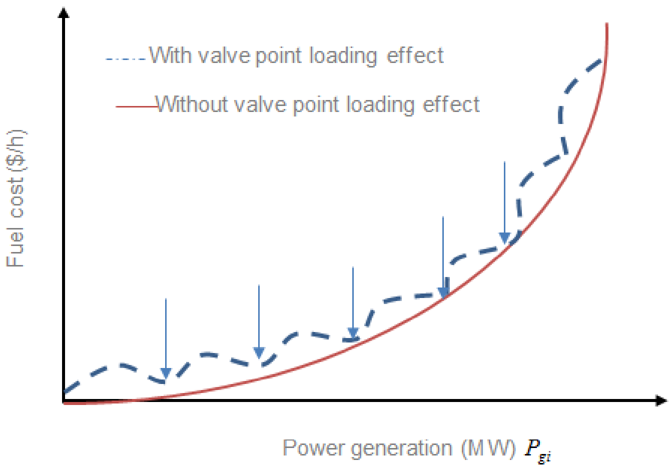



**Figure 2.3:** **Conventional cost curve with/without VPLE**(Ghasemi et al., 2016)

**Equality constraints**

(a) Real power balance constraints

The generated power of wind and thermal units is equal to power demand plus line losses and can be represented mathematically as in (2.4)(Ghasemi et al., 2016).

$$\sum_{i=1}^{N_G} P_{gi} + \sum_{k=1}^{N_W} P_{wh_{j,k}} = P_D + P_{Loss} \qquad \text{[MW]} \qquad (2.4)$$

where

$P_D$ Total load demand of the power system

$P_{Loss}$ Transmission line losses of the power system

In electric power transmission systems, losses are generated due to transmission over long distances. Thus, relative to low-load-density areas, across vast spaces, and in wind-thermal dynamic economic dispatch modeling, it is important to consider transmission losses. According to (Huang et al., 2018)The formula for transmission loss includes the term defined by the first quadratic term of equation (2.3), later referred to as George's formula. This is referred to as the loss formula or B-coefficient method. In addition, a linear term and a constant are added to the quadratic to provide a more accurate representation of transmission losses. According to (Huang et al., 2018), This is known as Kron's formula and is shown in equation (2.5) (Jiang et al., 2015). The B-coefficient method retains losses that could be avoided if each plant output were better scheduled based on the expected economic appeal to the projected load.

$$P_L = \sum_{m=1}^{N_G+N_W} \sum_{m'=1}^{N_G+N_W} P_m B_{mm'} P_{m'} + \sum_{m=1}^{Ng+Nw} B_{om} P_m + B_{00} \quad \text{[MW]} \qquad (2.5)$$

where

$m$ and $m'$ Wind-thermal generator capacity

$B_{mm'}$,' $B_{0m}$ and $B_{00}$ Loss of generation factors

The objective function expressed in (2.1) is subjected to the following inequality constraints.

**Inequality Constraints**

(a) Generation capacitor upper limits

The supply power of each generating unit in operation should not surpass the upper and lower limits of an operational unit, as stated in Equation (2.6).

$$\begin{aligned} &P_{gi}^{\min} \le P_{gi} \le P_{gi}^{\max}, i = 1,2,...N_G \\ &0 \le P_{j,h} \le P_{R,j,h}, j = 1,2,...N_w \end{aligned} \qquad (2.6)$$

where

$P_{gi}^{\max}$ Maximum output power of the $i^{th}$ thermal generating unit

$P_{gi}^{\min}$ Minimum output power of the $i^{th}$ thermal generating unit

$P_{R,j,h}$ Rated power of $j^{th}$ wind output generator

(b)Ramping up /down constraints

All online units have a limit to their operating range due to the ramping rate of an individual unit (Kothari & Dhillon, 2012). The generator can operate in three regions: steady-state operation, unit increasing its power output, and unit decreasing its power output. These regions can be represented diagrammatically by Figure 2.4.

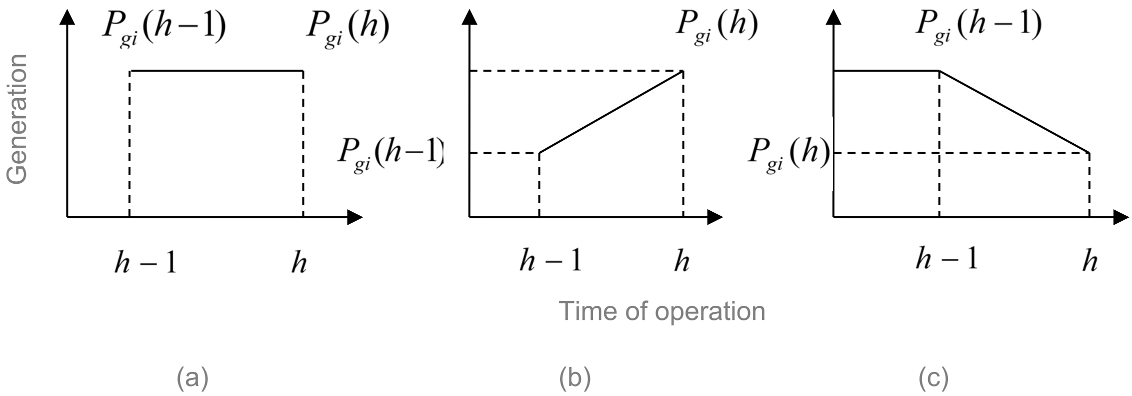



**Figure 2.4:** **Ramp rate limits of a generating unit** (Kothari & Dhillon, 2012)

Figure 2.4(a) represents steady state operation. Figure 2.4(b) shows the unit increasing its power output. Figure 2.4(c) depicts the unit decreasing power output. The operating boundaries are defined by Equation (2.7)

$$\begin{cases} -R_i^{\,D} \le (P_{gi,h} - P_{gi,h-1}) \le R_i^{\,U}, \text{if } P_{gi,h-1} \ge P_{gi}^{\ \min} \\ R_i^{\,0} \le \left|P_{gi,h} - P_{gi,h-1}\right| \le R_i^{\,1}, \text{if } 0 < P_{gi,h-1} < P_{gi}^{\ \min} \end{cases} \qquad (2.7)$$

Where

$R_i^{\,D}$ Minimum ramping down rate

$R_i^{\,U}$ Maximum ramping up rate

$R_i^{\,0}$ Lower limit of the variation rate

$R_i^{1}$ Upper limit of variation rate

(c) Minimum on/off constraints

These constraints denote the minimum time the generating unit must be committed once it starts up, and the minimum time it must be de-committed once it is shut down.

The minimum on/off constraints are presented as given in equation (2.8)

$$\begin{cases} [T^{R}{}_{i,h-1} - T^{R}{}_{\min}][s_{i,h-1} - s_{i,h}] \ge 0 \\ [T^{S}{}_{i,h-1} - T^{S}{}_{\min}][s_{i,h} - s_{i,h-1}] \ge 0 \end{cases} \qquad (2.8)$$

Where,

$T^{R}{}_{i,h-1}$ Continuous online time

$T^{S}{}_{i,h-1}$ Continuous Offline time,

$T^{R}{}_{\min}$ Minimum online time

$T^{S}{}_{\min}$ Minimum offline time

$s_{i,h}$ Operating status of the $i^{th}$ thermal generator

(d) Prohibited operating zone

The prohibited operating zones of the input-output characteristic curve for a conventional generator are as follows: vibration of the shaft bearing due to the steam valve, or due to machine malfunction, or malfunction of other ancillary components of the generator. Moreover, it's almost impossible to determine the real input-output characteristics curve near a prohibited zone from actual input-output testing or operational data (Kothari & Dhillon, 2012). These conditions are the ones in Equation (2.9) and are depicted in Figure 2.5.

$$\left\{\begin{array}{l} \{P_{gi}^{\min} \le P_{gi} \le P_{gi,1}^{L} \quad (i=1,2...,N_G) \\ \{P_{gi,j-1}^{U} \le P_{gi} \le P_{gi,j}^{L} \quad (j=2,3...,Nz_i) \text{ and } (i=1,2...,N_G) \\ \{P_{gi,Nz_i}^{U} \le P_{gi} \le P_{gi}^{\max} \quad (i=1,2...,N_G) \end{array}\right\} \tag{2.9}$$

Where

$Nz_i$ Number of prohibited zones of $i^{th}$ thermal generating unit

$P_{gi,j}^{L}$ Lower bound of the $j^{th}$ prohibited zone of the $i^{th}$ thermal generating unit

$P_{gi,j-1}^{U}$ Upper bound of $j^{th}$ prohibited zone of the $i^{th}$ thermal generating unit

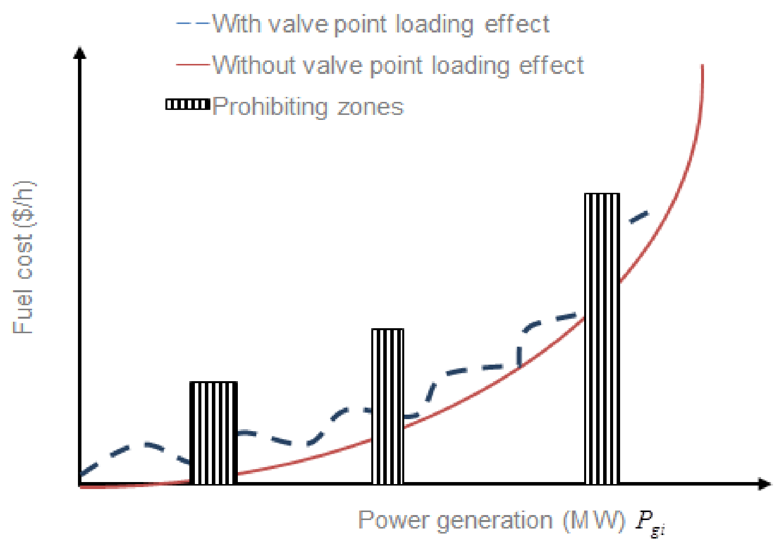



**Figure 2.5**: **Effective cost attributes with restricted operational zones (Kothari & Dhillon, 2012)**

(f) Wind Power constraints

The total power available from a wind turbine is equal to the product of the mass flow rate $\frac{dm}{dt}$, air density $\rho$, speed $v$ and the blades swept the cross-sectional area A (Masters, 2004). Therefore, the wind turbine's output power is given by Equation (2.10).

$$P_W = \frac{\rho A v^3}{2} \qquad [\text{ MW}] \qquad (2.10)$$

The power curve shown in Figure 2.5 is the power the wind turbine will produce as a function of turbine height and wind speed. Thus, the data related to this curve will dictate how much output power will be generated by this wind turbine and how much wind energy will be produced. The turbine curve shown in Figure 2.6, however, comprises three performance regions: the cut-in wind speed, the rated wind speed, and the cut-out/furling wind speed.

where

| | | |
|---|---|---|
| $V_C$ | Cut in wind speed | [m/s] |
| $V_R$ | Rated wind speed | [m/s] |
| $V_F$ | Cut out or Furling wind speed | [m/s] |

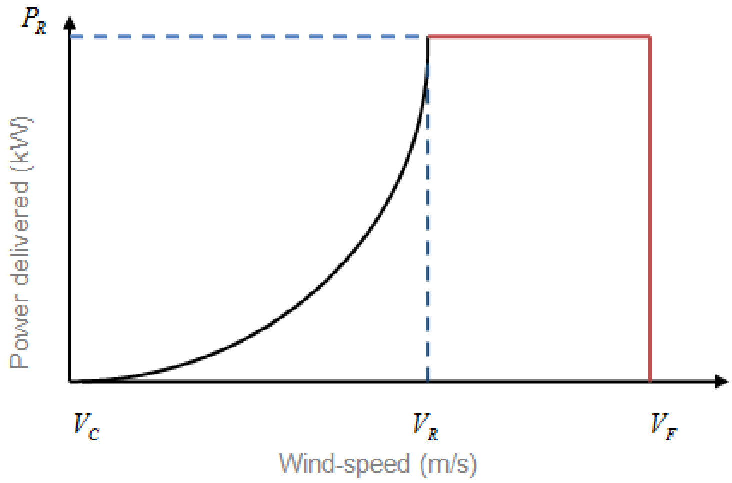



**Figure 2.6: A power curve for a general wind turbine** (Masters, 2004)

The wind generation output power as a function of wind speed is represented through a simplified linear piecewise function, which is given in equation (2.11) (Qu et al., 2016)

$$\begin{cases} P_w = 0, & v < v_C \text{ or } v > v_F \\ P_w = P_R \dfrac{v - v_C}{v_r - v_C}, & v_C \le v \le v_R \\ P_w = P_R, & v_R \le v \le v_F \end{cases} \qquad [\text{MW}] \qquad (2.11)$$

where

| | | |
|---|---|---|
| $P_w$ | Power delivered by a wind turbine | [MW] |
| $P_R$ | Rated power | [MW] |
| $V_C$ | Cut in wind speed | [m/s] |
| $V_R$ | Rated wind speed | [m/s] |
| $V_F$ | Cut out or Furling wind speed | [m/s] |

**(i) Wind power probability density function (PDF)**

Representing short-term wind speeds is important for assessing wind energy potential. PDFs are commonly used to assess wind speed. The applicability of various PDFs has been assessed in several regions worldwide. Since wind power is calculated as a simple function of wind speed distribution parameters, choosing a probability density function is important to the study of wind energy.

Using a PDF that better represents the wind speed data reduces the approximation error associated with uncertain wind power predictions. The Weibull two-parameter PDF and the Rayleigh PDF are the two most common PDFs when assessing wind speed data. In the literature, the Weibull PDF has been used to model wind speed data to solve the wind-thermal economic dispatch problem. The Weibull PDF retains several advantages over other probability distributions. For its many advantages, it requires only the estimation of two parameters and has a positively skewed probability density function that favours average wind speeds. Historically, it has proven a good fit for real-world wind speed data. If the Shape and Scale parameters are known at one height, a procedure exists to determine their equivalents at another height.

The Rayleigh PDF is a one-parameter distribution, a special case of the two-parameter Weibull PDF. A shape parameter of 2 can also be used. Many PDFs have been used in the literature to represent wind speed conditions, but for solving the economic dispatch problem of wind intermixed with gas units, the Weibull PDF is most commonly used.

The wind speed at a specific location can be assessed as a Probability Density Function (Masters, 2004) and is illustrated in Figure 2.7. The probability density function is given by $f(v)$ , where is defined as the wind speed. The area under the curve equals one, and this area under the curve equals the probability that the wind speed is equal to between $v_1$ and. $v_2$

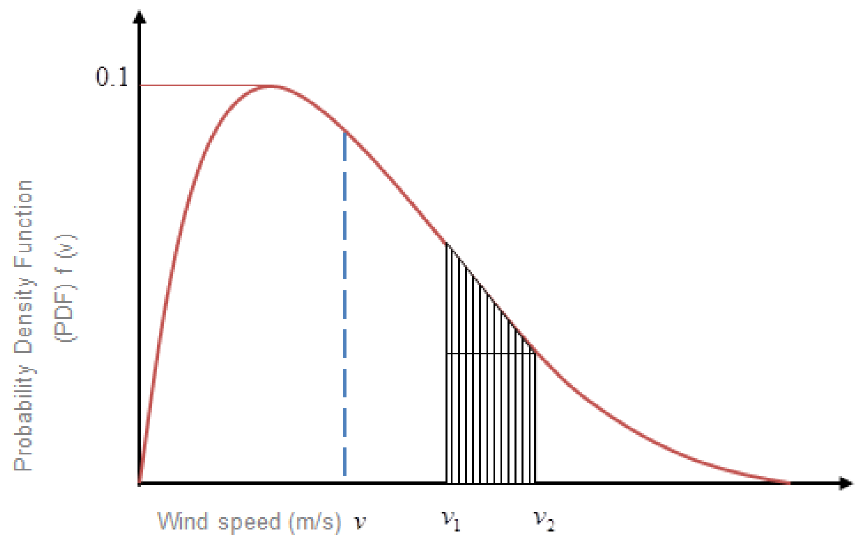



**Figure 2.7: Wind velocity Probability Density Function (PDF)** (Masters, 2004)

The PDF can be represented mathematically as given in Equations (2.12) and (2.13) respectively (Masters, 2004).

$$P(v_1 \leq v \leq v_2) = \int_{v_1}^{v_2} f(v)dv \qquad (2.12)$$

$$P(0 \leq v \leq \infty) = \int_{0}^{\infty} f(v)dv = 1 \qquad (2.13)$$

The PDF can be represented using the Weibull Probability Density Function (wpdf) (Masters, 2004) and is given in Equation (2.14).

$$f(V) = \frac{k}{c}\left(\frac{V}{c}\right)^{k-1} e^{-(V/c)^k} \quad 0 \leq V < \infty \qquad (2.14)$$

Where

$c$ Scale parameter at a given location [m/s]

$k$ Shape parameter at a given location (dimensionless).

### (ii) Cumulative Weibull distribution function (cdf)

The probability that the wind speed is less than some specified wind speed $V$ and is given in Equation (2.15), and it is called a cumulative distribution function(Masters, 2004) $P(v \leq V) = F(V) = \int_{0}^{V} f(v)dv$ (2.15)

Where

$V$ Wind speed is a random variable

Wind speed

The cumulative distribution function of wind speed is given by Equation (2.16).

$$F(V) = P(v \le V) = \int_0^V \left( \frac{k}{c} \left( \frac{V}{c} \right)^{k-1} e^{-(V/c)^k} \right) dv \quad (2.16)$$

Where

To solve Equation (2.16), let's assume the new integration variable $x = \left( \frac{V}{c} \right)^k$ .The limits of the integral $V = 0$ and $V = x$ , become $x = 0$ and $x = \left( \frac{V}{c} \right)^k$ . Differentiating the definition of $x = \left( \frac{V}{c} \right)^k$ with respect $V$ result to $dx = \frac{k}{c} \left( \frac{V}{c} \right)^{k-1} dV = \frac{kV^{k-1}}{c^k} dV$ and $dV = \frac{c^k}{kV^{k-1}} dx$ . .Substituting the definition of $x$ , the new limits, and the formula for $dV$ into Equation (2.16). Finally, all mathematical representation and resultant cumulative wind probability density function is given in Equation (2.17)

$$F(V) = P(v \le V) = \int_{V=0}^{V=x} \left( \frac{k}{c} \left( \frac{V}{c} \right)^{k-1} e^{-(V/c)^k} \right) dv = \int_{x=0}^{x=\left(\frac{V}{c}\right)^k} \left( \frac{kV^{k-1}}{c^k} e^{-x} \frac{c^k}{kV^{k-1}} \right) dx$$

$$= \int_0^{\left(\frac{V}{c}\right)^k} e^{-x} dx = -e^{-x} \Big|_0^{\left(\frac{V}{c}\right)^k} = 1 - \exp\left( -\left( \frac{V}{c} \right)^k \right) \quad (2.17)$$

If there is a probability that the wind speed exceeds a certain threshold, then the cumulative distribution function can be defined as in Equations (2.18) and (2.19).

$$P(v \ge V) = 1 - P(v \le V) = 1 - F(V) \quad (2.18)$$

$$P(v \ge V) = 1 - P(v \le V) = 1 - F(V) = 1 - \left[ 1 - \exp\left( -\left( \frac{V}{c} \right)^k \right) \right] = \exp\left( -\left( \frac{V}{c} \right)^k \right) \quad (2.19)$$

In several applications of the Weibull PDF, statistical moments are required (Altunkaynak et al., 2012). The computations for the mean, variance, and wind power can be simplified by a single integration over the distribution's moments. According to the PDF, the $n^{th}$ moment for any probability density function $f(V)$ is defined by Equation (2.20) (Leon-Garcia, 2008).

$$\overline{V^n} = \int_{V_{min}}^{V_{max}} V^n f(V) dV = \int_0^{\infty} V^n \frac{k}{c}\left(\frac{V}{c}\right)^{k-1} e^{-(V/c)^k} dV = k\int_0^{\infty} V^{n-1}\left(\frac{V}{c}\right)^k e^{-(V/c)^k} dV$$

$$= k\int_0^{\infty} V^{n-1} x e^{-x} \frac{c}{k} x^{\frac{1}{k}-1} dx = c\int_0^{\infty}\left(cx^{\frac{1}{k}}\right)^{n-1} x^{\frac{1}{k}} e^{-x} dx = c^n \int_0^{\infty} x^{\frac{n}{k}} e^{-x} dx = c^n \Gamma\left(\frac{n}{k}+1\right),\ n = 1,2...$$

(2.20)

Where

$\Gamma$ is the Gamma function and is given in Equation (2.21)

$$\Gamma(\beta) = \int_0^{\infty} x^{\beta-1} \exp(-x) dx \tag{2.21}$$

Equation (2.20) is obtained precisely the same way as Equation (2.17). The mean and variance of the Weibull function are given in Equations (2.22) and (2.23), respectively.

$$\mu_{Weibull} = \int_{V_{min}}^{V_{max}} V f(V) dV = c\Gamma\left(\frac{1}{k}+1\right) \tag{2.22}$$

$$(\sigma^2 + \mu^2) = \int_{V_{min}}^{V_{max}} V^2 f(V) dV = \int_0^{\infty} V^2 \frac{k}{c}\left(\frac{V}{c}\right)^{k-1} e^{-(V/c)^k} dV = \overline{V^2} = c^2\Gamma\left(\frac{2}{k}+1\right) \tag{2.23}$$

A continuous or discrete probability distribution defines the wind power curve. (Xian Liu & Xu, 2010). In a continuous probability function, the wind speed is between the cut-in and the rated wind speed. In a discrete probability function, the wind speed is between the rated and furling wind speed, and also where the output power has a constant zero value, i.e., below the cut-in and cut-out wind speed. The mathematical formulation of the continuous and discrete probability functions used to find the total output power of the wind generating units is described in the following section.

**(iii) Continuous probability**

The difference between continuous and discrete probability is that a constant probability distribution cannot be presented in tabular form; instead, a mathematical representation is used to illustrate a continuous PDF. The mathematical equation that describes continuous probability is called the probability density function and is given in Equation (2.14). It can also be called a density function, PDF, or a PDF. It possesses the following properties.

- The graphical representation of the probability density function is continuous over a particular range.
- The area circumscribed by the curve of the density function and the x-axis is equal to one when calculated over the domain of the variable.

- The probability that a random variable estimates a value between $v_1$ and $v_2$ is equal to the area under the density function bounded by $v_1$ and $v_2$.

It is estimated that wind speed has a given distribution, such as the Weibull distribution; it is necessary to transform that distribution into a wind power distribution. This conversion can be achieved as follows, with $V$ as the wind speed is a random variable and $P_W$ Is wind power a random variable? According to (Leon-Garcia, 2008)The linear transformation of a random variable $P_w$, such as the one presented in Equations(2.11) in the interval $v_C \le v \le v_R$ is given in Equation (2.24(Leon-Garcia, 2008)

$$P_w = T(V) = aV + b \qquad \text{[MW]} \qquad (2.24)$$

Where $\boldsymbol{a}$ is the constant greater than zero, and $V$ has CDF $F_V(v)$, Then we needed to find the CDF of the function $F_{P_w}(p_w)$. For simplicity, let $P_w = Y$ and $V = X$ .Then Equation (2.24) becomes $Y = aX + b$ and CDFs of $X$ and $Y$ be $F_X(x)$ and $F_Y(y)$. The event $\{Y \le y\}$ occurs when $A = \{aX + b \le y\}$ occurs. If $a > 0$, then $A = \{X \le (y-b)/a\}$ and If $a < 0$, then $A = \{X \ge (y-b)/a\}$ . Then the CDFs of $Y$ be represented by Equations (2.25).

$$\begin{aligned} F_Y(y) &= P\left[X \le \frac{y-b}{a}\right] = F_X\left[\frac{y-b}{a}\right], a > 0 \\ F_Y(y) &= P\left[X \ge \frac{y-b}{a}\right] = 1 - F_X\left[\frac{y-b}{a}\right], a < 0 \end{aligned} \qquad (2.25)$$

The PDF of $Y$ can be obtained by differentiating Equation (2.25) with respect to $y$ and using the chain rule for derivatives $dF/dy = dF/du \times du/dy$ .

Where

$\boldsymbol{u}$ Argument of $F$

After differentiation of Equation (2.25) for both $a > 0$ and $a < 0$**,** the PDF of the function $Y$ is given in Equations 2.26

$$\begin{aligned} f_Y(y) &= \frac{1}{a} f_X\left[\frac{y-b}{a}\right], a > 0 \\ f_Y(y) &= \frac{1}{-a} f_X\left[\frac{y-b}{a}\right], a < 0 \end{aligned} \qquad (2.26)$$

The results in Equation (2.26) can be combined using one equation and is given in Equation (2.27).

$$f_Y(y) = \left|\frac{1}{a}\right| f_X\left[\frac{y-b}{a}\right] \tag{2.27}$$

Using Equation (2.24), and (2.27), the Weibull pdf of the wind power output random variable in the continuous range can be found and is given in Equation (2.28). Firstly, the constants $a$ and $b$ are represented by the following Equations.

$$a = \frac{P_R}{v_R - v_C}, b = \frac{-P_R v_C}{v_R - v_C}$$

The equations for $a$ and $b$ can be read directly from Figure 2.6 between cut-in and rated wind speed. The workout for the equation of the continuous range is as follows

$$\left|\frac{1}{a}\right| f_V\left[\frac{w-b}{a}\right] = \frac{1}{\frac{P_R}{v_R - v_C}} f_V\left[\frac{w + \frac{P_R v_C}{v_R - v_C}}{\frac{P_R}{v_R - v_C}}\right]$$

$$\left|\frac{1}{a}\right| f_V\left[\frac{w-b}{a}\right] = \frac{v_R - v_C}{P_R} f_V\left[\frac{\frac{w(v_R - v_C) + P_R v_C}{v_R - v_C}}{\frac{P_R}{v_R - v_C}}\right]$$

$$= \frac{v_R - v_C}{P_R} f_V\left[\frac{w(v_R - v_C) + P_R v_C}{P_R}\right] = \frac{h v_C}{P_R} f_V\left[\frac{hw}{P_R} + 1\right] v_C$$

$$h = \frac{v_R - v_C}{v_C}$$

Using the Weibull distribution function in Equation (2.14), the PDF in the continuous range is:

$$f_{P_w}(p_w) = \frac{khv_C}{P_R c}\left[\frac{\left(1 + \frac{hw}{P_R}\right) v_C}{c}\right]^{k-1} \exp\left\{-\left[\left(\frac{\left(1 + \frac{hw}{P_R}\right) v_C}{c}\right)^k\right]\right\}, v_C \le v \le v_R \tag{2.28}$$

**(iv) Discrete probabilities**

A discrete random variable is a random variable that takes values from a countable set, that is, a finite set $S_F = \{f_1, f_2, f_3 ... f_n\}$ (Leon-Garcia, 2008). A simple example that clearly describes the concept of a discrete random variable is a Coin toss. A coin is tossed two times, and the appearance of heads and tails is monitored. The probabilities of the toss are defined by the following set $S = \{TT, TH, HT, \text{and } HH\}$ , where $T$ and $H$ represents tails and heads, respectively. Let $F$ be the number of tails in the two tosses, and allocate each outcome in $S$ from the $S_F = \{0, 1, 2\}$ . The four outcomes and corresponding values $F$ are monitored. Therefore, the discrete random variable takes on values in the set $S_F = \{0, 1, 2\}$ . The discrete portion of the wind power output random variable must account for the piecewise-linear properties shown in Equation (2.11). The discrete probability is on intervals where $P_w = 0$ and $P_w P_R$. The mathematical representation of the two events $P_w = 0$ and $P_w = P_R$ are given in Equations (2.29) and (2.30) separately (Hetzer et al., 2008)

$$\begin{aligned}
&\Pr(P_w = 0) = \Pr(V < v_C) + \Pr(V \geq v_F) \\
&= 1 - \exp\left[-\left(v_C / c\right)^k\right] + 1 - \Pr(V < v_F) = 1 - \exp\left[-\left(v_C / c\right)^k\right] + 1 - F(V) \\
&= 1 - \exp\left[-\left(v_C / c\right)^k\right] + 1 - \left[1 - \exp\left[-\left(v_F / c\right)^k\right]\right] \\
&= 1 - \exp\left[-\left(v_C / c\right)^k\right] + \exp\left[-\left(v_F / c\right)^k\right]
\end{aligned} \tag{2.29}$$

$$\begin{aligned}
&\Pr(P_w = P_R) = \Pr(v_R \leq V \leq v_F) \\
&= \Pr(V \leq v_F) - \Pr(V \geq v_R) \\
&= 1 - \exp\left[-\left(v_F / c\right)^k\right] - \left[1 - \exp\left[-\left(v_R / c\right)^k\right]\right] \\
&= \exp\left[-\left(v_R / c\right)^k\right] - \exp\left[-\left(v_F / c\right)^k\right]
\end{aligned} \tag{2.30}$$

**(v) Total cumulative distribution function of wind power**

The Cumulative Distribution Function (CDF) of wind power is represented by both continuous and discrete probabilities, and is given in equation (2.28) (Xian Liu, 2012).The solution of Equation (2.28) follows several axioms in probability theory (Leon-Garcia, 2008). The axiom allows us to view events as objects that possess a property with attributes similar to those of physical mass. There are three axioms in probability theory, namely, Axiom I, Axiom II, Axiom III, and Axiom IV. Axiom I states that the probability is nonnegative; Axiom II states that there is a fixed total amount of probability, namely 1 unit. Axiom III states that the probability of two disjoint events is the sum of their individual probabilities. Axiom III' is the special case of Axiom III, and is used to handle experiments with infinite sample spaces. These laws are shown mathematically and are given in Equation (2.31)(Leon-Garcia, 2008)

$$\begin{array}{ll} Axiom\ I & 0 \le P[A] \\ Axiom\ II & P[S] = 1 \\ Axiom\ III & \text{If } A \cap B = \emptyset, \text{then } P[A \cup B] = P[A] + P[B] \\ Axiom\ III' & \text{If } A_1, A_2 ... \text{is a sequence of events such that} \\ & A_i \cap A_j \text{ for all } i \ne j, \text{then} \end{array} \quad (2.31)$$

$$P\left[\bigcup_{k=1}^{\infty} A_k\right] = \sum_{k=1}^{\infty} P[A_k]$$

Where

$P[A]$, and $P[B]$ Probability of a random number $A$ and $B$

$S$ Sample space.

The random variable $P_w$ is a piecewise linear function of $V$, and several cases need to be considered based on the axioms, laws, and are given in Equations (2.32 ) to (2.36)( Liu et al., 2021).

Case I

$$\begin{aligned} &P_w = P_R \\ &\Pr(P_w \le p_w) = \Pr(P_w \le P_R) = 1 \end{aligned} \quad (2.32)$$

Case II

$$\begin{aligned} &P_w > P_R \\ &\Pr(P_w \le p_w) = \Pr(P_w < 0) + \Pr(P_R < P_w < p_w) \\ &= \Pr(P_w \le P_R) + 0 = 1 \end{aligned} \quad (2.33)$$

Case III

$$\begin{aligned} &P_w < 0 \\ &\Pr(P_w \le p_w) = \Pr(P_w < 0) - \Pr(p_w < P_w < 0) \\ &\Pr(P_w \le p_w) = 0 - 0 = 0 \end{aligned} \quad (2.34)$$

Case IV

$$\begin{aligned} &P_w = 0 \\ &\Pr(P_w \le p_w) = \Pr(P_w \le 0) = \Pr(P_w < 0) + \Pr(P_w = 0) \\ &= 0 + \Pr(V \le v_C) + \Pr(V > v_F) = F_V(v_C) + 1 - F_V(v_F) \end{aligned} \quad (2.35)$$

Case V

$$\begin{aligned} &0 < P_w < P_R \\ &\Pr(P_w \le p_w) = \Pr(P_w \le 0) = \Pr(P_w < 0) + \Pr(0 < P_w < p_w) \\ &= F_V(v_C) + 1 - F_V(v_F) + \Pr(0 < P_w < p_w) \\ &\text{Where} \\ &\Pr(0 < P_w < p_w) = \Pr[v_C < V < f_V(P_w)^{-1}] \\ &= F_V(P_w)^{-1} - F_V(v_C) \end{aligned} \quad (2.36)$$

By substituting Equation (2.28) to Equation (2.36), the resultant expression is given in

Equation(2.36).

$$Pr(\ 0 < P_w < p_w) = 1 - exp\left\{-\left[\left(\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right] - F_V(v_C)\right. \tag{2.37}$$

Lastly, substituting Equation (2.37) into Equation (2.36) yields the combined equation in Equation (2.38).

$$Pr(\ P_w \le p_w) = 1 - exp\left\{-\left[\left(\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right] - F_V(v_F)\right. \tag{2.38}$$

By taking the inverse of $F_V(v_F)$ result with the expression given in Equation (2.39). This equation denotes the CDF of the output energy random character $P_w$.

$$\begin{cases} 0 & \mathrm{P} < 0 \\ 1 & \mathrm{P} \ge \mathrm{P_R} \\ 1 - \exp\left\{-\left[\left(1 + \frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right] + \exp\left\{-\left[\frac{v_F}{c}\right]^k\right\}\right. ;0 \le P < P_R \end{cases} \tag{2.39}$$

Since wind speed is a stochastic variable at any given time, the system operator may overestimate or underestimate the actual capacity of the wind-generating unit. Due to the nonlinearity of the wind-generating system, it is convenient to model for underestimation and overestimation in the cost curve. These important concepts are discussed separately in the following sections.

**(vi) Overestimation**

It occurs when the actual output wind power from the wind power system is less than the predicted output, so the system operator must purchase power from another power generation source. The overestimation of wind power can be represented mathematically and is given in Equation (2.40) (Hetzer et al., 2008)

$$C_{r,w,i}(P_{w_i} - P_{W_{i,av}}) = k_{r,i}(P_{w_i} - P_{W_{i,av}}) = k_{r,i}\int_0^{w_i}(P_{w_i} - P_W)f_{P_W}(w)dw \tag{2.40}$$

Where

$k_{r,i}$ Reserve cost coefficient for the $i^{th}$ wind power generator.

$P_{w_i}$ Predicted wind power

$P_{W_{i,av}}$ Available/Generated random wind power

$f_{P_W}(w)dw$ Probability distribution function for a Random variable $P_W$

Equation (2.40) is solved through the wind power probability Equations (2.28), (2.29), and (2.30). The analysis of the overestimation of wind power output is presented in the following way, and an analytical expression is given in Equation (2.41)(Liu & Xu, 2010a). For simplicity, the presentation is focused on a particular wind energy conversion system, and actual wind power is only denoted by $P_W$ and leave subscript $i, av$. The predicted wind power output for the first turbine is denoted by $P_{w_1}$. The overestimated wind power output is $\phi_{OE} = P_{w_i} - P_W$ and the rated power represented as $P_R$. The average of the overestimated wind power output is given by $\phi_{OE} = s_1 + s_2$( Liu & Xu, 2010a).

Where

$$s_1 = w_1 P(W = 0) = w_1\left\{1 - \exp\left[-\left(\frac{v_{in}}{c}\right)^k\right] + \exp\left[-\left(\frac{v_{out}}{c}\right)^k\right]\right\} \quad (2.41)$$

$$\begin{aligned} s_2 &= \int_0^{w_r} \max(w_1 - w, 0) f_W(w)dw \\ &= \int_0^{w_1} (w_1 - w) f_W(w)dw \\ &= w_1 \int_0^{w_1} f_W(w)dw - \int_0^{w_1} w f_W(w)dw \\ &= w_1[F_W(w_1) - F_W(0)] - H_{31} \end{aligned} \quad (2.42)$$

The solution of the first term can be easily found using Equation (2.16) and $H_{31}$ is given in Equation (2.43).

$$\begin{aligned} H_{31} &= \int_0^{w_1} w f_W(w)dw \\ &= \frac{khv_C}{P_R c}\int_0^{w_1} w\left[\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right]^{k-1} \exp\left\{-\left[\left(\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right]\right\}dw \end{aligned} \quad (2.43)$$

Equation (2.43) is expressed in terms of integrals, which makes it very inconvenient to execute and analyse. The expression is broken down using Equation (2.16) and the substitution of variables such as $t = (1 + hw/w_r)v_{in}$. The equation is solved as follows.

Knowing that $t = (1 + hw / w_r) v_{in}$, the variable $w$ can be easily found to be $w = (tw_r - v_{in} w_r)/hv_{in}$, $dw = w_r / hv_{in} dt$ and $y = (t/c)$. Substituting these variables to Equation (2.43), leads to the following expression as given in Equation (2.44).

$$
\begin{aligned}
&= \frac{khv_{in}}{w_r c} \int_0^{w_1} \left( \frac{tw_r - v_{in} w_r}{hv_{in}} \right) \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] \frac{w_r}{hv_{in}} dt \right. \\
&= \frac{khv_{in}}{w_r c} \left( \frac{tw_r}{hv_{in}} \right) \left( \frac{w_r}{hv_{in}} \right) \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. \\
&- \frac{khv_{in}}{w_r c} \left( \frac{v_{in} w_r}{hv_{in}} \right) \left( \frac{w_r}{hv_{in}} \right) \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. \\
&= \frac{khv_{in}}{w_r c} \left( \frac{t(w_r)^2}{(hv_{in})^2} \right) \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. \\
&- \frac{khv_{in}}{w_r c} \left( \frac{v_{in}(w_r)^2}{(hv_{in})^2} \right) \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. \qquad (2.44) \\
&= \frac{k}{c} \left( \frac{tw_r}{hv_{in}} \right) \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. - \frac{kw_r}{hc} \int_0^{w_1} \left[ \frac{t}{c} \right]^{k-1} \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. \\
&= \left( \frac{kw_r}{hv_{in}} \right) \int_0^{w_1} t \left[ \frac{t^{k-1}}{c^k} \right] \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right. - \frac{kw_r}{h} \int_0^{w_1} \left[ \frac{t^{k-1}}{c^k} \right] \exp\left\{ - \left[ \left( \frac{t}{c} \right)^k \right] dt \right.
\end{aligned}
$$

$$
\text{Knowing that } y = \left( \frac{t}{c} \right)^k \therefore dy = \frac{kt^{k-1}}{c^k} dt, t = cy^{\frac{1}{k}} \text{ and } dt = \frac{c^k}{kt^{k-1}} dy \therefore
$$

$$
\begin{aligned}
&= \left( \frac{kw_r}{hv_{in}} \right) \int_0^{w_1} cy^{\frac{1}{k}} \left[ \frac{t^{k-1}}{c^k} \right] \exp(-y) \left( \frac{c^k}{kt^{k-1}} dy \right) - \frac{kw_r}{h} \int_0^{w_1} \left[ \frac{t^{k-1}}{c^k} \right] \exp(-y) \left( \frac{c^k}{kt^{k-1}} dy \right) \\
&= \frac{w_r c}{hv_{in}} \int_{v_{in}}^{v_1} y^{\frac{1}{k}} \exp(-y) dy - \frac{w_r}{h} \int_{v_{in}}^{v_1} \exp(-y) dy, \text{ where } v_1 = \frac{v_{in} + (v_r - v_{in}) w_1}{w_r}
\end{aligned}
$$

According to Equations (2.16) and 2.21, Equation (2.44) can be fully solved and is given in Equation (2.45).

$$
\begin{aligned}
= \frac{w_r c}{hv_{in}} \Gamma \left[ 1 + \frac{1}{k}, \left( \frac{v_{in}}{c} \right)^k \right] - \frac{w_r c}{hv_{in}} \Gamma \left[ 1 + \frac{1}{k}, \left( \frac{v_1}{c} \right)^k \right] + \\
+ \frac{w_r}{h} \left[ exp \left( - \frac{v_{in}}{c} \right)^k - \left[ exp \left( - \frac{v_1}{c} \right)^k \right] \right] \qquad (2.45)
\end{aligned}
$$

The combination of Equations (2.41), (2.42), and (2.45) represents the mathematical model of overestimation of wind power output and is given in Equation (2.46).

$$E(Y_{oe,j,t}) = w_{j,t}\left[1-\exp\left(-\frac{v_{C\,j,t}}{c_{j,t}}\right)^{k_{j,t}} + \exp\left(-\frac{v_{F\,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] +$$

$$+\left(\frac{P_{R,j,t}v_{C,j,t}}{v_{R,j,t}-v_{C,j,t}} + w_{j,t}\right)\left[\exp\left(-\frac{v_{C\,j,t}}{c_{j,t}}\right)^{k_{j,t}} - \exp\left(-\frac{v_{1\,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] + \tag{2.46}$$

$$+\left(\frac{P_{R,j,t}c_{j,t}}{v_{R,j,t}-v_{C,j,t}}\right)\left[\Gamma\left(1+\frac{1}{k_{j,t}},\left(\frac{v_{1,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right) - \Gamma\left(1+\frac{1}{k_{j,t}},\left(\frac{v_{R,j,t}}{c^{k,j,t}}\right)^{k_{j,t}}\right)\right]$$

**(vii) Underestimation**

The principle of underestimation is the opposite of overestimation; it occurs when the actual wind output power exceeds the forecast, so the excess is wasted, and the operator must reimburse the wind power output. It can be represented by the following mathematical expression and is given in Equation (2.47) (Hetzer et al., 2008)

$$C_{p,w,i}(P_{W_{i,av}} - P_{w_i}) = k_{p,i}(P_{W_{i,av}} - P_{w_i}) = k_{p,i}\int_0^{w_i}(P_W - P_{w_i})f_{P_W}(w)dw \tag{2.47}$$

Where

$k_{p,i}$ Reserve cost coefficient for the $i^{th}$ Wind power generator.

The analyses of underestimation are done the same way as those of the overestimation. The average of the underestimated wind power output is given by

$\varphi_{UE} = s_1 + s_2$

Where

$$s_1 = w_r P(W = w_r)$$
$$= w_r\left\{\exp\left[-\left(\frac{v_r}{c}\right)^k\right] - \exp\left[-\left(\frac{v_{out}}{c}\right)^k\right]\right\} \tag{2.48}$$

$$s_2 = \int_{v_r}^{v_{out}} \max(w - w_1) f_W(w) dw$$

$$= \int_{0}^{w_1} (w - w_1) f_W(w) dw$$

$$= \int_{v_r}^{v_{out}} w f_W(w) dw - w_1 \int_{v_r}^{v_{out}} f_W(w) dw$$

$$\int_{v_r}^{v_{out}} w f_W(w) dw - w_1 \left[ \int_{v_r}^{v_{out}} f_W(w) dw + \int_{v_r}^{v_1} f_W(w) dw \right]$$

$$= H_{32} - w_1 [F_W(v_{out}) - F_W(v_r)] + w_1 [F_W(v_1) - F_W(v_r)]$$

$$= H_{32} - w_1 \left[ \exp\left[ -\left(\frac{v_r}{c}\right)^k \right] - \exp\left[ -\left(\frac{v_{out}}{c}\right)^k \right] \right] + w_1 \left[ \exp\left[ -\left(\frac{v_r}{c}\right)^k \right] - \exp\left[ -\left(\frac{v_1}{c}\right)^k \right] \right] \tag{2.49}$$

The derivation of $H_{32}$ is exactly the same as $H_{31}$which is derived in Equation (2.44), but the difference is the limits of the definite integral. The integral for $H_{32}$ is from $v_R \le v \le v_1$ .The expression for $H_{32}$ is given in Equation (2.50).

$$\frac{w_r c}{h v_{in}} \Gamma\left[ 1 + \frac{1}{k}, \left(\frac{v_1}{c}\right)^k \right] - \frac{w_r c}{h v_{in}} \Gamma\left[ 1 + \frac{1}{k}, \left(\frac{v_r}{c}\right)^k \right] + + \frac{w_r}{h} \left[ \exp\left( -\frac{v_R}{c} \right)^k - \left[ \exp\left( -\frac{v_1}{c} \right)^k \right] \right] \tag{2.50}$$

Then, the combination of Equations (2.44), (2.48), (2.49), and (2.50) result in the following expression as presented in Equation (2.51).

$$E(Y_{ue,j,t}) = (P_{R,j,t} - w_{j,t}) \left[ \exp\left( -\frac{v_{R,j,t}}{c_{j,t}} \right)^{k_{j,t}} - \exp\left( -\frac{v_{F,j,t}}{c_{j,t}} \right)^{k_{j,t}} \right] + + \left( \frac{P_{R,j,t} v_{C,j,t}}{v_{R,j,t} - v_{C,j,t}} + w_{j,t} \right) \left[ \exp\left( -\frac{v_{R,j,t}}{c_{j,t}} \right)^{k_{j,t}} - \exp\left( -\frac{v_{1,j,t}}{c_{j,t}} \right)^{k_{j,t}} \right] + + \left( \frac{P_{R,j,t} c_{j,t}}{v_{R,j,t} - v_{C,j,t}} \right) \left[ \Gamma\left( 1 + \frac{1}{k_{j,t}}, \left(\frac{v_{1,j,t}}{c_{j,t}}\right)^{k_{j,t}} \right) - \Gamma\left( 1 + \frac{1}{k_{j,t}}, \left(\frac{v_{R,j,t}}{c^{k,j,t}}\right)^{k_{j,t}} \right) \right] \tag{2.51}$$

### 2.2.1.2 Multi-criterion economic dispatch.

Atmospheric pollution has increased tremendously due to industrial advancement. When coal is used in the thermal plant utilities, harmful gaseous pollutants such as carbon oxides($CO_2$),, sulphur oxides($SO_2$), and nitrogen oxides($NO_x$). The Green Power committee suggests using low-emission sources to address the economic dispatch problem. Therefore, there is a need for improved control techniques that can achieve lower pollution levels at a realistic fuel cost. The emission gases, such as $CO_2$, $SO_2$ and $NO_x$, and polluted by thermal generators, should be included in the optimisation objective function. Wind turbine generators are not subject to pollution, and the output power of a wind plant depends on wind speed, so the emission cost for the wind generator is zero. The overall emission of these pollutants is the sum of the quadratic cost function and the emission values, with the exponential function of the valve-point loading effect. This can be represented mathematically as (Abdelaziz et al., 2016)

$$E_T = \sum_{i=1}^{N_E} E_i(P_i) = \sum_{i=1}^{N_E} (\alpha_i + \beta_i P_i + \gamma_i P_i^2 + \lambda_i \times \exp(\varepsilon_i \times P_i)) \text{ [kg/h]} \quad (2.52)$$

Where

$E_T$ Total emission [kg/h]

$\alpha_i, \beta_i, \gamma_i,$ Emission coefficients of the generating unit I in [kg/h], [kg/MWh], and [ton/MW$^2$h]

$\varepsilon_i, \lambda_i$ Valve point loading effect emission coefficients of the generating unit i

The pollutants $CO_2$, $SO_2$ and $NO_x$ The conventional generator can be formulated as given in Equations (2.53) to (2.55) respectively.

$$E_{CO_2 i} = \sum_{i=1}^{n} (\alpha_{CO_2 i} + \beta_{CO_2 i} P_{gi} + \gamma_{CO_2 i} P_{gi}^2 + \lambda_{CO_2 i} \times \exp(\varepsilon_{CO_2 i} \times P_{gi})) \text{ [kg/h]} \quad (2.53)$$

$$E_{SO_2 i} = \sum_{i=1}^{n} \alpha_{SO_2 i} + \beta_{SO_2 i} P_{gi} + \gamma_{SO_2 i} P_{gi}^2 + \lambda_{SO_2 i} \times \exp(\varepsilon_{SO_2 i} \times P_{gi})) \text{ [kg/h]} \quad (2.54)$$

$$E_{NO_x i} = \sum_{i=1}^{n} \alpha_{NO_x i} + \beta_{NO_x i} P_{gi} + \gamma_{NO_x i} P_{gi}^2 + \lambda_{NO_x i} \times \exp(\varepsilon_{NO_x i} \times P_{gi})) \text{ [kg/h]} \quad (2.55)$$

Where

$\alpha_{CO_2 i}$, $\beta_{CO_2 i}$ $\gamma_{CO_2 i}$ $\lambda_{CO_2 i}$ and $\varepsilon_{CO_2 i}$ $CO_2$ emission coefficients of $i^{th}$ thermal generator in [kg/h], [kg/MWh], [kg/MW$^2$h], [kg/h], and [kg/MWh]

$\alpha_{SO_2 i}$, $\beta_{SO_2 i}$ $\gamma_{SO_2 i}$ $\lambda_{SO_2 i}$ and $\varepsilon_{SO_2 i}$ $SO_2$ emission coefficients of $i^{th}$ thermal generator in [kg/h], [kg/MWh], [kg/MW$^2$h], [kg/h], and [kg/MWh]

$\alpha_{NO_x i}$, $\beta_{NO_x i}$ $\gamma_{NO_x i}$ $\lambda_{NO_x i}$ and $\varepsilon_{NO_x i}$ $NO_x$ emission coefficients of $i^{th}$ thermal generator in [kg/h], [kg/MWh], [kg/MW$^2$h], [kg/h], and [kg/MWh]

Therefore, the total emission is the sum of the three emissions and is given in Equation (2.56):

$$E_T = E_{CO_2 i} + E_{SO_2 i} + E_{NO_x i} \qquad \text{[kg/h]} \qquad (2.56)$$

The fuel cost function of the wind-thermal dispatch problem is given in Equation(2.2), and the emission functions of $CO_2$, $SO_2$ and $NO_x$ gases are given in Equations (2.53)-(2.55) individually. To formulate the multi-objective functions into a single objective function by using the price penalty factors given in Equations (2.57) to (2.59), respectively(Abdelaziz et al., 2016; Bhattacharya & Chattopadhyay, 2010; Jiang et al., 2015).

$$F_{CO_2} = \sum_{i=1}^{N_E} \left( \begin{array}{l} a_i + b_i P_{gi} + c_i P_{gi}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi}) \right) \right| + \\ h_{CO_2 i} \times (\alpha_{CO_2 i} + \beta_{CO_2 i} P_{gi} + \gamma_{CO_2 i} P_{gi}^2 + \lambda_{CO_2 i} \times \exp(\varepsilon_{CO_2 i} \times P_{gi}) \end{array} \right) \qquad \text{[\$/h]} \qquad (2.57)$$

$$F_{SO_2} = \sum_{i=1}^{N_E} \left( \begin{array}{l} a_i + b_i P_{gi} + c_i P_{gi}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi}) \right) \right| + \\ h_{SO_2 i} \times (\alpha_{SO_2 i} + \beta_{SO_2 i} P_{gi} + \gamma_{SO_2 i} P_{gi}^2 + \lambda_{SO_2 i} \times \exp(\varepsilon_{SO_2 i} \times P_{gi}) \end{array} \right) \qquad \text{[\$/h]} \qquad (2.58)$$

$$F_{NO_x} = \sum_{i=1}^{N_E} \left( \begin{array}{l} a_i + b_i P_{gi} + c_i P_{gi}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi}) \right) \right| + \\ h_{NO_x i} \times (\alpha_{NO_x i} + \beta_{NO_x i} P_{gi} + \gamma_{NO_x i} P_{gi}^2 + \lambda_{NO_x i} \times \exp(\varepsilon_{NO_x i} \times P_{gi}) \end{array} \right) \qquad \text{[\$/h]} \qquad (2.59)$$

Where

$F_{CO_2}$ CEED fuel cost of $CO_2$ emission

$F_{SO_2}$ CEED fuel cost of $SO_2$ emission

$F_{NO_x}$ CEED fuel cost of $NO_x$ emission

The total fitness function is given in Equation (2.60) by considering $CO_2$, $SO_2$ and $NO_x$ emissions.

$$F_T = \sum_{i=1}^{N_E} \left( \begin{array}{l} a_i + b_i P_{gi} + c_i P_{gi}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi}) \right) \right| + \\ h_{CO_2 i} \times (\alpha_{CO_2 i} + \beta_{CO_2 i} P_{gi} + \gamma_{CO_2 i} P_{gi}^2 + \lambda_{CO_2 i} \times \exp(\varepsilon_{CO_2 i} \times P_{gi}) + \\ h_{SO_2 i} \times (\alpha_{SO_2 i} + \beta_{SO_2 i} P_{gi} + \gamma_{SO_2 i} P_{gi}^2 + \lambda_{SO_2 i} \times \exp(\varepsilon_{SO_2 i} \times P_{gi}) + \\ h_{NO_x i} \times (\alpha_{NO_x i} + \beta_{NO_x i} P_{gi} + \gamma_{NO_x i} P_{gi}^2 + \lambda_{NO_x i} \times \exp(\varepsilon_{NO_x i} \times P_{gi}) \end{array} \right) \qquad \text{[\$/h]} \qquad (2.60)$$

The PPFs for $CO_2$,$SO_2$ and $NO_x$ are calculated using a ratio of quadratic fuel cost to the emission and are given in Equations (2.61) to (2.63)

$$h_{CO_2 i} = \frac{a_i + b_i P_{gi\max} + c_i P_{gi\max}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi\max}) \right) \right|}{(\alpha_{CO_2 i} + \beta_{CO_2 i} P_{gi\max} + \gamma_{CO_2 i} P_{gi\max}^2 + \lambda_{CO_2 i} \times \exp(\varepsilon_{CO_2 i} \times P_{gi\max})} \qquad \text{[\$/kg]} \qquad (2.61)$$

$$h_{SO_2i} = \frac{a_i + b_i P_{gi\max} + c_i P_{gi\max}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi\max}) \right) \right|}{(\alpha_{SO_2i} + \beta_{SO_2i} P_{gi\max} + \gamma_{SO_2i} P_{gi\max}^2 + \lambda_{SO_2i} \times \exp(\varepsilon_{SO_2i} \times P_{gi\max})} \quad [\$/kg] \quad (2.62)$$

$$h_{NO_xi} = \frac{a_i + b_i P_{gi\max} + c_i P_{gi\max}^2 + \left| e_i \sin\left( f_i (P_{gi}^{\min} - P_{gi\max}) \right) \right|}{(\alpha_{NO_xi} + \beta_{NO_xi} P_{gi\max} + \gamma_{NO_xi} P_{gi\max}^2 + \lambda_{NO_xi} \times \exp(\varepsilon_{NO_xi} \times P_{gi\max})} \quad [\$/kg] \quad (2.63)$$

Where

$h_{CO_2i}$ Max-Max price penalty factor of $CO_{2i}$emission

$h_{SO_2i}$ Max-Max price penalty factor of $SO_{2i}$emission

$h_{NO_xi}$ Max-Max price penalty factor of $NO_{xi}$emission

### 2.2.2 Study of optimisation techniques employed in solving the single-area dynamic wind-thermal economic dispatch problem

#### 2.2.2.1 Single criteria

This section investigates different optimisation methods for solving the wind-thermal economic dispatch problem to optimally control a single-area power system, accounting for the dynamic behaviour of renewable energy sources. Several optimisation methods have been used to solve a dynamic economic dispatch problem, including classical and heuristic methods (Zhu, 2015). The conventional classical methods include Unconstrained Optimisation Approach(UCA), Nonlinear Programming(NLP), Linear Programming(LP), Quadratic Programming (QP), Generalised Reduced Gradient Method(GRGM), Newton Method(NM), Network Flow Programming(NFP), Mixed-Integer Programming(MIP), and Interior Point methods(IP). The heuristic search methods, such as Artificial Neural Network(ANN), Evolutionary Algorithms(EAs), Tabu Search(TS), Particle Swarm Optimisation(PSO), and Non-quantitative approaches that address uncertainties in objectives and constraints, such as Probabilistic Methods(PM), Fuzzy logic(FL), and Analytic Hierarchical Process(AHP).

In a power system including wind generating units, it is necessary to consider spinning reserves such as Up Spinning Reserve (USR) and Down Spinning Reserve (DSR). The Up Spinning Reserve (USR) is the reserve capacity for an unexpected increase in load, a random drop in wind turbine generator power output, or the required outage of thermal generators. The Down Spinning Reserve (DSR) is the reserve capacity designed for rapid load decreases and erratic increases in wind turbine generator power output. (Chen et al., 2006), presented a Direct Search Method (DSM) to solve the economic dispatch problem with wind integration. The objective function considered in their work included both conventional operation cost and wind turbine generation cost (WTGs) that are owned by the Independent Power Producer (IPP). Several optimisation constraints were considered, including minimum/maximum spinning reserves (up/down), power balance, and minimum/maximum capacity for both thermal and wind generators. The wind power is modelled using the Weibull Probability Density Function (WPDF). Also, Lee 2017 formulated the Optimal Spinning Reserve Wind-thermal (OSRWT) economic dispatch to determine the optimal spinning reserve (both USR and DSR) schedule for a wind-thermal power system that minimizes total social cost over a given period while satisfying system operating constraints. An efficient algorithm, Evolutionary Iteration Particle Swarm Optimization (EIPSO), is used to solve the OSRWT problem. This algorithm is a modification of PSO and Evolutionary Programming (EP). The Hybrid Optimization of Multiple Energy Resources (HOMER) software is used to simulate hourly wind speed data at a test site, and then generate the wind speed probability using the Weibull probability density function.

The Authors (Hetzer et al., 2008) and (Meyyappan & Pandu, 2015), investigated the penalty cost for overestimation and underestimation of the available wind power output, which are included in their objective cost function for the optimum dispatch problem. The MATLAB optimisation toolbox is used to evaluate the optimal solution of the overall cost function, subject to linear/nonlinear optimisation constraints. In addition to that, the Authors (Meyyappan & Pandu, 2015)Provided a brief explanation of how the Wavelet Neural Network (WNN) can be used as an alternative to the Probabilistic Neural Network (PNN), Artificial Neural Network (ANN), and General Regressive Neural Networks (GRNNs) for extracting wind forecast data, as shown in Figure 2.8. The wind-thermal dispatch optimisation problem is first solved using the Primal-Dual-Interior-Point (PDIP) method, and later using Differential Evolution (DE) and Bacterial Foraging Technology (BFT). The solution is compared between the two heuristic methods.

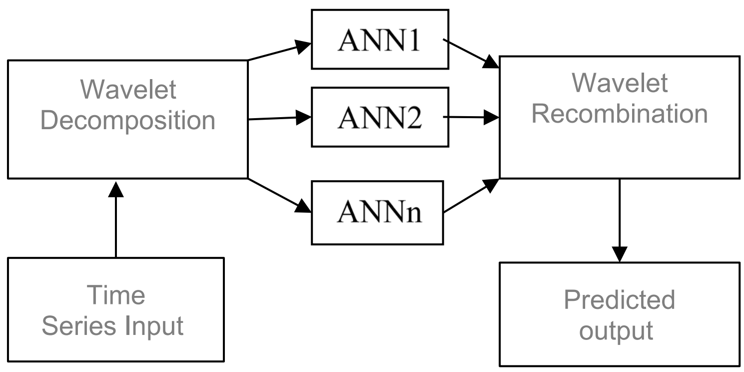



**Figure 2.8:** **Predicted model for WNN** (Meyyappan & Pandu, 2015)

The WNN (Wavelet Neural Network) forecast model in Figure 2.7 comprises three stages: pre-signal processing, signal forecasting, and post-signal processing. The time-series input signal is decomposed using wavelets, with the signals treated with multiple wavelet factors. The neural networks' input is the result of the wavelet decomposer. The neural network's output is processed by wavelet recombination, which enhances all signals, and the resulting output is the forecast. The Weighted Probabilistic Neural Network (WPNN) (Krishnasamy & Nanjundappan, 2016) is a more generalized version of the predictive PNN applied by Meyyappan & Pandu (with respect to data extraction efficiency). The cost function for fuel application considers multiple fuel-producing units with valve-point loading, subject to various power system limitations, and BBO + SQP is applied to obtain a better solution.

A new optimal method, the Genetic-Teaching Learning-Based Optimisation (G-TLBO), was introduced by Güçyetmez & Çam (2016) for global optimisation of the wind-thermal dispatch problem. The wind-thermal dispatch problem is defined as an optimal power flow problem applied to the cost function relative to system constraints, with wind-speed generation over- and under-estimations.

In the article by Lu et al. (2008), the valve-point loading quadratic cost function was researched. An Independent Power Producer (IPP) owns a wind farm with a great dispersion of wind turbines. The economic use of the IPP will significantly reduce generation costs, as the utility will purchase the required wind power. Wind power generation depends on wind speed, as shown in the wind power curve. The authors solve the economic dispatch problem using the Direct Search Method (DSM) and the Evolutionary Simulated Annealing (ESA) algorithms, where ESA is employed to find the global optimal solution, and DSM is used to improve the solution and reduce computing time. Authors (Boqiang & Chuanwen, 2009) assess an instantaneous economic dispatch model with wind power and a temporary economic dispatch model with wind power, where wind power is certain, for the economic emission generation system problem, using an optimization algorithm and the power city risk dispatch framework. According to Morales et al. (2009), their work created a dispatch problem that quantifies the reserves produced in a market-clearing scenario that accounts for wind uncertainty and represents the best spinning/non-spinning reserves of a robust system with substantial wind penetration. The limits are provided by the market for energy selling and first-stage variables (variables that are not impacted by scenario realization at each time step at each scenario, i.e. the start-up and shut-down scheduling for each generation unit), reality system operation, and second-stage variables (these variables, which are dependent on the specific scenarios and are connected to actual operation of the system i.e. wind power spilling). Constraints on market versus real system operation and on first-stage versus second-stage variables are taken into account. In Xian Liu & Xu’s work, wind economic dispatch is understood.

Therefore, in this research, the optimal solution to thermal dispatch problems uses probabilistic infeasibility prevention and the Lagrangian method to obtain optimal results. In Jiang et al. (2011), a quadratic cost function with valve-point loading effects was studied as a dynamic economic dispatch problem. Transmission losses and real power requirements are treated as equality constraints; however, bulk constraints for thermal and wind generating units, as well as up/down spinning reserves, are treated as inequality constraints in this study. This study requires 24-hour-ahead wind energy predictions. The dispatch problem is solved by the Improved Particle Swarm Optimization (IPSO) technique (X. Liu, 2010), which utilizes the Beta distribution to operate generation units forecasted at the time of the studied sharewind generation output. The issue is optimized for wind-thermal dispatch using a chance-constrained nonlinear programming (CCNP).

The authors ( Li & Jiang, 2011) measured an economic dispatch for 24 hours with a combined Risk administration, which involves in estimating the risk assessment, and this would benefit the system operator (SO) to vary a decent adjustment between revenues and risks, depending on system equality and inequality limitations. The particle swarm optimisation (PSO) heuristic is used to optimise the power system problem. Wind power output is viewed as a stochastic (random) variable by the authors (Zhou et al., 2011) and the reserve cost is accounted for in the total objective function. The dynamic economic dispatch problem is solved via a nonlinear Primal-Dual Interior-Point (PDIP) method. Constraints, such as risk-based spinning reserves, minimum and maximum wind power outputs, absolute power limits of the thermal plant, and ramp speed limits of conventional units, are considered. The authors (Reddy et al., 2013), formulated the economic dispatch fuel cost function with valve-point loading effect. The spinning reserve supplied by the thermal units is included in the cost function, along with the underestimation and overestimation of wind power and the direct cost of wind power. Covariance Matrix Adaptation with Evolution Strategy (CMA-ES) and Mean Learning Technique (MLT) are used to find solutions to the economic dispatch problem, which considers both. Inequality and equality constraints. The authors (Liang et al., 2015) formulated the economic dispatch problem, which includes both active and reactive power. The Firefly algorithm is used as a heuristic optimization approach. The authors (Li et al., 2013) presented a Lagrangian approach to solve a dynamic economic dispatch problem. The quasi-Newton method (QNM) is used to optimise the economic dispatch problem, subject to equality and inequality constraints. The review summary of the wind-thermal economic emission dispatch problem is given in Table 2.3.

#### 2.2.2.2 Multi-criteria

In recent years, research on multi-criteria optimisation methods has increased. Decisions with multiple criteria are predominant in government, military, industry, and other organisations (Kothari & Dhillon, 2012). Research from a wide range of fields, such as mathematics, management, science, economics, engineering, and others, has contributed to solution methods for multi-criteria optimisation problems. The multi-criteria optimisation problem is also called multi-performance, multi-objective, or vector optimisation. The engineer's goal is to maximize or minimize not a single objective function but numerous objective functions simultaneously. The purpose of multi-performance problems in the mathematical programming framework is to optimize multiple objective functions, subject to a set of system constraints.

(Kuo, 2010) and (Qu et al., 2016)Formulated the Economic Emission Dispatch (EED) problems with a stochastic wind power model. The dispatch problem is solved by using a Summation-based Multi-Objective Differential Evolution Algorithm (SMODE). The wind power output is modelled as a system constraint by using stochastic variables and applying probability theory. Ramp rate limits, prohibited zones, and transmission losses are treated as nonlinear constraints, while power balance is treated as a linear constraint. PSO and GA optimisation methods are used by (Kuo, 2010). The Summation-based Multi-Objective Differential Evolution (SMODE) and NSGAII are used as heuristic optimisation methods in (Qu et al., 2016). The authors (Xian Liu & Xu, 2010), solved the economic emission dispatch problem, including overestimation and underestimation constraints in the cost function. The impact of wind power is presented using the Weibull Distribution function. The authors (Bhattacharya & Chattopadhyay, 2010) presented a biogeographical optimization method to solve the Economic Emission Dispatch (EED) problem. The EED problem is formulated to minimize total emissions from conventional power-generating units.

The authors (Azizipanah-Abarghooee et al., 2015; Dubey et al., 2015; A. Ghasemi et al., 2016; Jadhav & Roy, 2013a; Mondal et al., 2013a; Peng et al., 2012a) examined the formulation of the economic dispatch problem, taking into account a quadratic cost function with valve-point loading effects. The Weibull probability distribution function (PDF) is used to model wind power. Depending on who owns the wind farm, the cost function also includes the reserve cost resulting from the unavailability of wind power (overestimation) and the penalty for underutilising available wind (underestimation).

Equation 2.27 provides a representation of the emission cost function in terms of the quadratic cost and an exponential function. Several optimisation techniques, such as the Modified Teaching-Learning Algorithm)(MLTA) (Azizipanah-Abarghooee et al., 2012), Bi-Population Chaotic Differential Evolution (BPCDE)(Peng et al., 2012b), Gbest

Guided Artificial Bee Colony(GABC) algorithm (Jadhav & Roy, 2013b), and Gravitational Search Algorithm(GSA)(Mondal et al., 2013b), the combination of Hybrid Flower Pollination(HFPA-TVFSM) Algorithm and Time Varying Fuzzy Selection Mechanism(Dubey et al., 2015), Gravitational Acceleration Enhanced Particle Swarm Optimization(GAEPSO) algorithm(Jiang et al., 2015) and Online Learning Honey Bee Mating Optimization(OLHBMO) (Ghasemi et al., 2016) are used to solve the wind-thermal economic dispatch problem subject to system linear and nonlinear constraints.

The authors(Chaudhary et al., 2020; Pandit, Chaudhary, et al., 2015a; Zhu et al., 2014) used a quadratic fuel cost to solve the Combined Emission Economic Dispatch (CEED) problem, omitting the cost function for overestimation and underestimation. The optimisation problem is resolved using a decomposition-based Multi-Objective Evolutionary Algorithm (MOEA/D). (SPSO-DE) Series Particle Swarm Optimization-Differential Evolution(Chaudhary et al., 2020; Pandit, Chaudhary, et al., 2015a; Zhu et al., 2014). The problem is broken down into several smaller problems. Subject to system restrictions on both wind and thermal generating units. The Weibull Probability Distribution Function (WPDF) is used as a modelling tool for wind generator output. To reduce the multi-objective optimisation to a single objective function, (Niu & Wei, 2013) developed an economic dispatch problem utilising the Karush-Kuhn-Tucker conditions. The constraint optimisation problem is solved by combining the multiplier approach with the particle swarm algorithm.

The authors (Alham et al., 2016) considered flexible distributed energy resources, such as an energy storage system (ESS) and Demand Site Management(DSM), in the objective function. Different ESS are explained in(Patel, 1999). The ESS, DSM, and minimum and maximum power output for both wind and thermal generating units are included in the nonlinear constraints. The GAMS software is utilised to solve this Dynamic Economic Emission Dispatch (DEED) problem.

(Chen et al., 2017)The multi-objective optimisation problem is formulated with wind, thermal, and short-term hydro power plants. The power balance is represented by equality constraints without losses, and the inequality constraints are represented by minimum and maximum power limits for wind, thermal, and hydro energy. The cost function compensates for both wind overestimation and underestimation. A Modified Gravitational Search Algorithm based on the Non-dominated Sorting Genetic Algorithm-III (MGSA-NSGA-III) is used to find the optimal solution. It is associated with other methods described in the reported survey. The process is validated with four cascaded hydropower plants, three conventional units, and two wind power plants.

In (Elattar, 2018), the bio-objective optimisation problem is formulated with wind, thermal, and solar power plants. The power balance is represented by equality constraints without losses, and the inequality constraints are minimum and maximum power limits for wind, thermal, and solar energy. A Modified Harmony Search (MHS) algorithm is used to find the minimum cost for fuel and emissions for the optimization problem. Three thermal generators and a wind-and-solar unit are used to validate the process efficiency. Two of the conventional generators are synchronous generators, while the third one is a combined heat and power generator.

In (Hu et al., 2019)The economic dispatch problem is considered a dynamic optimization problem in which the conventional ramp rate is included in the power balance constraints. Wind power is modelled using a PDF, with wind represented by a direct cost function, an overestimation cost function, and an underestimation cost function. The problem is solved using a hybrid optimization method that combines GA (Genetic Algorithm) with Sequential Quadratic Programming (SQP) to form GA-SQP. Also,(Zhang et al., 2019), presented a multi-optimisation method that minimises both fuel and emission cost, using a quadratic cost function with valve-point loading effects and equality and inequality constraints in the objective function. Wind power is modelled using a PDF, including direct, overestimation, and underestimation terms in the cost function. The Gravitational Particle Swarm Optimization Algorithm (GPSOA) is used as a hybrid optimization algorithm to solve the WTEED problem.

In (Behera et al., 2021)The multi-objective optimisation problem is formulated with wind, thermal, and solar power. The power balance is represented by equality constraints without losses, and the inequality constraints are represented by minimum and maximum power limits for wind, thermal, and solar energy. The overestimation and underestimation of wind and solar are included in the cost function. A Constriction Factor-Based Particle Swarm Optimization (CFBPSO) algorithm is used to find the optimal solution and is compared with other algorithms reported in the literature. The algorithm is validated across six bus systems: three conventional units with valve-point loading, two wind, and two solar power plants.

The economic emission dispatch problem is solved using a decision-making technique, the Technique for Order Preference by Similarity to an Ideal Solution (TOPSIS) (Jin et al., 2021). Transmission losses are incorporated into the optimisation bi-function; balance restrictions are expressed as equalities and inequalities; and the concern is represented by a quadratic cost function and valve-point loading.

A non-convex optimization problem is presented in (Salkuti, 2021), where transmission losses are included in the equality constraints and ramp rate and forbidden zones are treated as inequality constraints. Wind energy modeling is presented as a stochastic function using a PDF.

In (Yan et al., 2022)The economic dispatch problem is formulated with valve-point loading effects in the quadratic function, and power balance constraints are included as equality constraints. Electric vehicles are used as energy storage systems to smooth the intermittency of wind power. This is done by developing the interactive model between wind power and Electric vehicles. Wind power is modelled using a two-parameter PDF. A self-adaptive multiple-learning multi-objective harmony-search(SAMLHS) algorithm is proposed for solving this complex optimization problem.

(Nagarajan et al., 2022), presented a paper on the economic dispatch problem that includes wind, thermal, and solar units. The economic dispatch is solved using the Improved Mayfly (IM) Optimization Algorithm. Different scenarios were considered: conventional units with solar, thermal units with wind, thermal-only units, and all combinations to validate the algorithm's effectiveness.

A non-convex optimization problem is formulated with wind and solar power (Azeem et al., 2023). The bio-objective function is transformed to a single one with PPF. A heuristic algorithm, the Salp Swarm Algorithm, is used to solve the optimization problem. This nonlinear optimization problem involves ramp rates of thermal units in inequality constraints and a loss term in equality constraints. Thereafter, the algorithm is tested against the latest survey. A combined heat and power(CHP) system is simulated with solar generation and wind generation with the global nondominated sorting genetic algorithm II (GNSGA-II(Zou et al., 2023). The following constraints, including power generation limits, heat generation limits, CHP unit capacity limits, power balances, heat balances, ramp rate limits, and spinning reserve, are considered in the optimization concern. Different scenarios are presented to illustrate the algorithm's effectiveness.

In the work of (Lalhmachhuana et al., 2024)Optimisation is performed with and without wind power integration; the power balance is treated as an equality, whereas the maximum and minimum powers of wind and thermal sources are treated as inequality constraints. Multi-Objective Particle Swarm optimisation (MOPSO) and Multi-Objective Ant-Lion optimisation (MOALO) are used, and the results for fuel and emissions are later compared.

In the work of (Secui et al., 2024b), Modified Social Group Optimization (MSGO) is used to verify the effectiveness of the optimization algorithm. The algorithm is further tested on an IEEE30 bus system that consists of 10-unit and 40-unit systems: two systems, each with 550MW, supply wind power injection. The equality and inequality constraints are incorporated into the objective function as power balance constraints and minimum and maximum limits for the wind and thermal generating units. Overestimation and underestimation of wind power are treated as penalty coefficients.

(Qiao et al., 2025) formulated the DEED problem with wind power, and an Electric Vehicle is used as a storage system to compensate for wind uncertainties. The intermittence between wind power and Electric vehicles is modelled using the conditional value-at-risk (CVaR) of IWEv. Constraints such as the power balance, the residual power of the EVs, travel constraints for EV owners, charging and discharging power, and climbing rate are included in DEEDR-IWEv. A self-adaptive multi-mode teaching-learning-based optimization (SaMmTLBO) algorithm is used to solve the DEED problem. The algorithm's feasibility is tested on a 10-unit system across different scenarios.

(Xiong et al., 2025) presented an EED problem that includes wind, solar, and small hydro runoff. The problem is solved using a Multi-Objective Artificial Gorilla Troops(MOAGT) optimizer. The algorithm is validated using the IEEE 30- and IEEE 18-bus systems. The bus allocation for RES differs across test systems. Recently, in (Meng et al., 2025), studied a day-ahead economic dispatch framework for microgrids with wind integration, combining battery storage and hybrid demand response (DR). The system is benchmarked across the following cases: conventional, wind-only, wind+storage, wind+DR, and fully integrated. The algorithm demonstrated that a fully integrated method can reduce cost, renewable utilization, and grid stability.

In the study of (Li et al., 2024), an economic scheduling prototype for combined energy units with stochastic wind and PV uncertainty. This paper efficiently utilizes electrolyzer thermal energy, incorporates carbon trading, and demand response. Information Gap Decision Theory (IGDT)-based methods manage risk while reducing operating cost and emissions.

(Mao et al., 2026)presented an integrated energy system scheduling framework that considers dynamic carbon trading and efficiency improvements. Optimizes multi-energy resources with carbon pricing to reduce operational cost and emissions. The problem is solved using Distributed Robust Optimization (DRO)-Model Predictive Control (MPC).

The review summary of the problem for wind-thermal economic dispatch is given in Table 2.1. Figure 2.9 illustrates the graphical representation of Table 2.1. This figure shows the number of publications per year. The review summary of single-area, single- and multi-criteria wind-thermal economic dispatch is presented in Table 2.3.

**Table 2.1:** **Quantity of papers for wind-thermal economic dispatch on a yearly basis**

| **Quantity of papers year-wise** | | |
|---|---|---|
| **Reference** | **Year of publication** | **Number of publications** |
| (George.,1945) | 1945 | 1 |

| | | |
|---|---|---|
| (Kron.,1951) | 1951 | 1 |
| (Gernald & Aaron.,1958) | 1958 | 1 |
| ( Schlueter et al, 1983) | 1983 | 1 |
| ( Schlueter et al, 1985) | 1985 | 1 |
| (Masters., 2004) | 2004 | 1 |
| (Miranda,and Pun Sio Hang, 2005) | 2005 | 1 |
| (Chen et al 2006), (Denny, and O'malley, 2006), (Patel., 2006) | 2006 | 3 |
| ( Lee,. 2007), (Ummels et al 2007), (Ruey-Hsun Liang, & Jian-Hao Liao, 2007),( Methaprayoon, et al, 2007), (Balamurugan & Subramanian ., 2007) | 2007 | 5 |
| (Hetzer et al 2008), (Lu et al., 2008), (Garcia., 2008) | 2008 | 3 |
| (Boqiang, and, Chuanwen, 2009), (Pappala et al, 2009), (Morales, et al 2009) | 2009 | 3 |
| (Kuo, 2010), (Yongpinget al, 2010), ( Liu. 2010), (Liu, and Xu. 2010), Liu and Xu 2010), (Da silva, 2010), (Kothari & Dhillon.,2010), )Bhattacharya & Chattopadhyay.,2010), Zhou et al.,2010) | 2010 | 10 |
| (Zhou, et al 2011), (Liu, 2011), (Jiang et al, 2011), (Makarov et al, 2011), (Constantinescu et al, (2011), (Meibom, et al 2011), (Li,& Jiang. 2011), (Hafiz, et al 2011), | 2011 | 8 |
| (Azizipanah-Abarghooee, et al 2012), (Peng, et al 2012 (Liang, et al 2012), ( Ciornei, & Kyriakides,. 2012), (LIU,et al. 2012), (Altunkaynak et al., 2012) | 2012 | 6 |
| (Adhav, and Roy .,2013), (Reddy et al., 2013), (Zhao, et al., 2013), (Mondal, et al., 2013), (Wang, et al., 2013), (Hafiz, et al., 2013), (Zhou et al.,2013), (Niu & Wei., 2013), (Krishnamurthy., 2013), (Abul'Wafa, 2013) | 2013 | 10 |
| (Zhu, et al .,2014), (Moeini-Aghtaie, et al., 2014), (Cheng & Zhang.,2014) | 2014 | 3 |
| (Geetha, et al., 2015), (Dubey, et al 2015), (Han, et al., 2015), (Osório, et al., 2015), (Pandit, et al 2015), (Liang, et al., 2015), (Jiang, et al., 2015), (Hedayati-Mehdiabadi, et al., 2015),(Meyyappan and Pandu, 2015), (Güçyetmez, & Çam, 2016), | 2015 | 10 |
| (Ghasemi, et al., 2016), (Krishnasamy, & Nanjundappan, 2016), (Lujano-Rojas, et al 2016), (Fu, et al., 2016), (Li, et al., 2016), (Zhou, et al., 2016), (Alham et al, 2016), (Qu et al, 2016), (Abdelaziz et al., 2016), (Patel., 2016), (Velamuri et al.,2016) | 2016 | 10 |
| (Chen et al.,2017) | 2017 | 1 |
| (Elatter 2018) | 2018 | 1 |
| (Hu et al., 2019), (Jiang et al., 2019) | 2019 | 2 |
| (Behera et al., 2020) | 2020 | 1 |
| (Jin et al., 2021), (Salkuti., 2021) | 2021 | 2 |
| (Yan et al., 2022), (Nagarajan et al.,2022) | 2022 | 2 |
| ( Azeem et al.,2023), (Zou et al., 2023). | 2023 | 2 |
| (Lalhmachhuana et al.,2024), (Secui et al. 2024) | 2024 | 2 |
| (Qiao et al., 2025), (Xiong et al.,2025), (Meng et al., 2025) | 2025 | 3 |

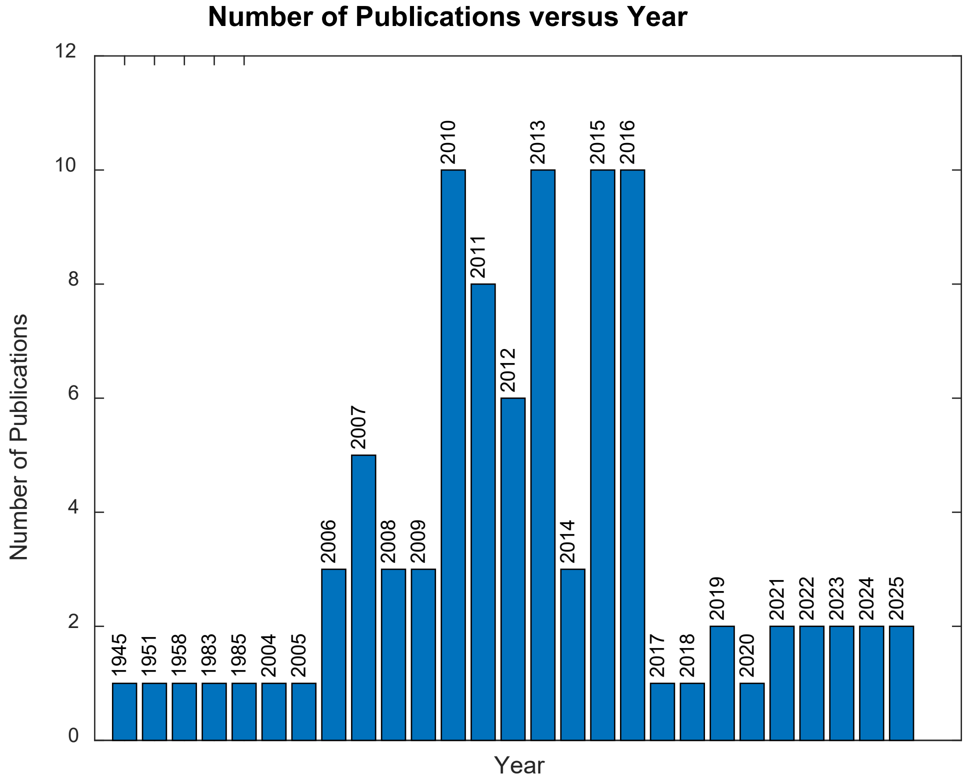



**Figure 2.9:** Quantity of papers for wind-thermal economic dispatch on yearly basis

**Table 2.2:** Techniques and algorithms applied to solve the single-area wind-thermal economic dispatch problem

| Most used methods and algorithms | | Reference paper | No. of publications |
|---|---|---|---|
| **Algorithm** | **Abbreviation** | | |
| Artificial Neural Networks | ANN | ( Methaprayoon, et al, 2007) | 1 |
| Biogeography-Based Optimization. | BBO | Bhattacharya & Chattopadhyay.,2010) | 1 |
| Bi-Population Chaotic Differential Evolution algorithm | BPCDE | (Peng, et al 2012) | 1 |
| Covariant Matrix Adaptation with Evolution Strategy with Mean Learning Technique | CMA-ES/ MLT | (Reddy et al., 2013) | 1 |
| Diffusion Particle Optimization | DPO | (Han, et al., 2015 | 1 |
| Direct Search Method | DSM | (Chen et al 2006), Alham et al, 2016) | 2 |
| Energy Storage System | ESS | (Alham et al, 2016) | 1 |
| Evolutionary Iteration Particle Swarm Optimisation | EIPSO | ( Lee,. 2007), | 1 |
| Evolutionary Simulated Annealing | ESA | (Lu et al., 2008) | 1 |

| | | | |
|---|---|---|---|
| Feasible Operation Region | FOR | (Hafiz, et al 2011), (Hafiz, et al 2013) | 2 |
| FireFly Algorithm | FFA | Liang, et al., 2015 | 1 |
| Flower Pollination Algorithm | FPA | (Velamuri et al,2016), (Abdelaziz et al., 2016) | 2 |
| Flying-Blick Method | FBM | (Makarov et al, 2011) | 1 |
| Forecasted Approach | FA | (Denny, and O'malley, 2006), (Constantinescu et al, (2011) | 2 |
| Fuzzy Logic | FL | (Miranda,& Pun Sio Hang, 2005), (Liang et al., 2015) (Jian-Hao Liao, 2007), (Geetha, et al., 2015) | 4 |
| Gbest guided Artificial Bee Colony algorithm | GABC | (Adhav, and Roy .,2013) | 1 |
| Genetic Algorithm –Ant Colony Approach | GA-API | ( Ciornei, & Kyriakides,. 2012) | 1 |
| Genetic Teaching Learning-Based Optimization | (G-TLBO) | (Güçyetmez, & Çam, 2015) | 1 |
| Gravitational Acceleration Enhanced Particle Swarm Optimization algorithm | GAEPSO | Jiang, et al., 2015 | 1 |
| Gravitational Search Algorithm (GSA) | GSA | (Mondal, et al., 2013), | 1 |
| Group Search Optimizer with Multiple Producers | GSOMP | (Li, et al., 2016), | 1 |
| Hybrid Flower Pollination Algorithm (HFPA)- Time Varying Fuzzy Selection Mechanism (TVFSM) | HFPA-TVFSM | (Dubey, et al 2015 | 1 |
| Hybrid Particle Swarm Optimization | HPSO | (Niu & Wei, 2013) | 1 |
| Interior Point Method | IPM | (Zhou, et al 2011), (Fu, et al., 2016) | 2 |
| Lagrangian Relaxation | LR | (Li, 2013),(Niu, & Wei, 2013) | 2 |
| Markov chain forecast model | MCFM | (Hedayati-Mehdiabadi, et al., 2015), | 1 |
| Mixed Interger | MI | (Meibom, et al 2011) | 1 |
| Modified Teaching-Learning Algorithm(MTLA) | MTLA | (Azizipanah-Abarghooee, et al 2012) | 1 |
| Multi-Objective Evolutionary Algorithm based on Decomposition | MOEA/D, | Zhu, et al .,2014 | 1 |
| Multi-Agent Genetic Algorithm | MAGA | (Moeini-Aghtaie, et al., 2014) | 1 |
| Non-dominated Sorting Genetic Algorithms | NSGA | (Abul'Wafa, 2013) | 1 |
| Online-Learning Honey Bee Mating Optimization | OLHBMO | (Ghasemi, et al., 2016), | 1 |

| | | | |
|---|---|---|---|
| Optimality Decomposition Coordination | ODC | (Zhou, et al., 2016) | 1 |
| Particle Swarm Optimisation | PSO | (Pappala et al, 2009), (Kuo, 2010), (Jiang et al, 2011), (Li,& Jiang. 2011), (Wang et al.,2013),(Salkuti.,2021) | 6 |
| Probabilistic Approach | PA | (Osório, et al., 2015), (Lujano-Rojas, et al 2016),( Da Silva., 2010), (Liu, 2011) | 4 |
| Probabilistic Kernel Density Forecasting | PKDF | Zhou et al.,2013 | 1 |
| Series Particle Swarm Optimization-Differential Evolution | (SPSO-DE) | Pandit, et al 2015 | 1 |
| Singular Weighted Method | SWM | (Yongping et al., 2010) | 1 |
| Stochastic Programming | SP | (Morales, et al 2009) | 1 |
| Summation based Multi-Objective Differential Evolution algorithm | SMODE | (Qu et al, 2016) | 1 |
| Weighted Probabilistic Neural Network | WPNN | (Krishnasamy, & Nanjundappan, 2016), | 1 |
| Weibull Probability Density Function | WPDF | (Hetzer et al 2008), (Liu, & Xu. 2010), (,Xu & Liu. 2010), ( Liu. 2010), | 4 |
| Modified Gravitational Search Algorithm based on the Non-dominated Sorting Genetic Algorithm-III (MGSA-NSGA-III) | MGSA-NSGA-III | (Chen et al.2017) | 1 |
| Modified Harmony Search (MHS) | MHS | (Elatter.,2018) | 1 |
| Genetic-Sequential Quadratic Programming | GA-SQP | (Hu et al., 2019) | 1 |
| Gravitational Particle Swarm Optimization Algorithm (GPSOA) | GPSOA | (Jiang et al., 2019) | 1 |
| Constriction Factor-Based Particle Swarm Optimization (CFBPSO) | CFBPSO | Behera et al., (2020) | 1 |
| Technique for Order Preference by Similarity to an Ideal Solution(TOPSIS) | TOPSIS | (Jin et al., 2021) | 1 |
| Self-Adaptive Multi-Learning Harmony-Search Algorithm (SAMLHS) | SAMLHS | (Yan et al., 2022) | 1 |
| Improved Mayfly (IM) Algorithm | IM | (Nagarajan et al.,2022) | 1 |
| Chaotic Salp Swarm Algorithm(CISSA) | CISSA | ( Azeem et al.,2023) | 1 |
| Global Nondominated Sorting Genetic Algorithm II (GNSGA-II) | GNSGA-II | (Zou et al., 2023) | 1 |
| Multi-Objective Particle Swarm optimisation(MOPSO) and Multi-Objective Ant-Lion optimisation(MOALO) | MOPSO and MOALO | (Lalhmachhuana et al.,2024) | 1 |
| Modified Social Group Optimization (MSGO) | MSGO | (Secui et al.,2024) | 1 |
| Self-adaptive Multi-mode Teaching-Learning-Based | SaMmTLBO | (Qiao et al., 2025) | 1 |

| | | | |
|---|---|---|---|
| Optimization (SaMmTLBO) algorithm | | | |
| Multi-Objective Artificial Gorilla Troops(MOAGT) optimizer | MOAGT | (Xiong et al.,2025) | 1 |
| Combining battery storage and hybrid demand response (DR) | DR | (Meng et al.,2025) | 1 |
| Information Gap Decision Theory(IGDT) | IGDT | (Li et al., 2024) | 1 |
| Distributed Robust optimization (DRO)-Model Predictive Control (MPC). | (DRO)-(MPC) | (Mao et al., 2026) | 1 |

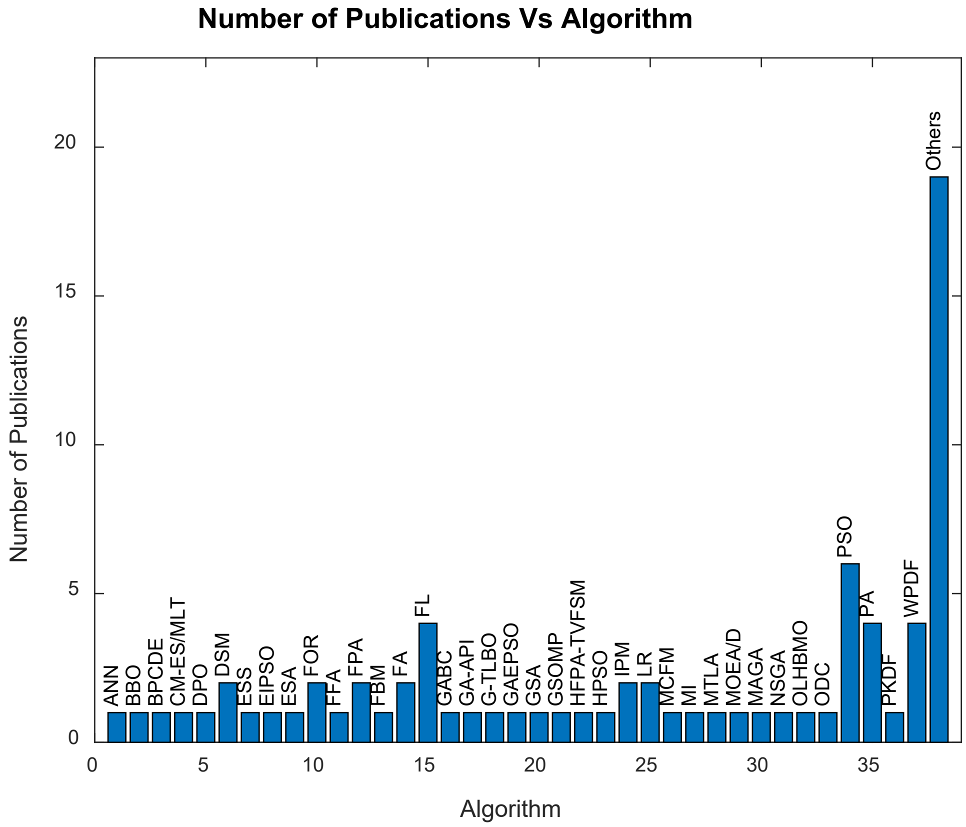



**Figure 2.10:** **Quantity of papers vs algorithm of wind-thermal economic dispatch**

**Table 2.3:** **Study of the literature on wind-thermal economic emission dispatch problem with single and multi-criteria**

| Reference Paper | System Components | | | Objective Function | | Constraints | | Algorithm | Software | Power System Considered | Real life implementation |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Wind | Storage | Coal, Diesel & Other | Single Objective | Multi Objective | Equality Constraints | Inequality Constraints | | | | |
| (Miranda,and Pun Sio Hang, 2005) | Yes | Nil | yes | Minimize fuel cost | Nil | Real power balance constraints | Minimum and maximum thermal technical limits for generators. Minimum wind power and maximum available wind power( predicted wind power) | Fuzzy logic | Not Specified | Not Specified | No |
| (Chen et al., 2006) | Yes | Nil | yes | Minimize fuel cost | Nil | Real power balance constraints. Generators' maximum up/ down spinning reserves. Thermal plant up/down spinning reserve contribution constraints. | Unit capacity constraints. Wind power curve constraints. System maximum ramping capability constraint | Direct search Method | FORTRAN | The Taiwan power system has 52 generating units. | No |
| (Denny, and O'malley, 2006) | Yes | Pumped storage | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Wind power forecast errors and load forecast errors constraints. system reliability criteria, and forced outage probabilities constraints | Forecast Approach | Not mentioned | Not specified | No |
| (Methaprayoonet al, 2007) | Yes | Nil | Nil | Fuel cost reduction | Nil | Nil | Nil | Artificial Neural Network | MATLAB | Wind farm in Lawton City that consists of 45 wind turbines, each rated at 1.65 MW | No |
| (Tsung-Ying Lee. 2007) | Yes | Nil | Yes | Operation and outage cost | Nil | Real power balance constraints. | Generator limits constraints. Min and Max wind turbine output power. | Evolutionary Iteration Particle Swarm | HOMER | Two practical power systems are used as numerical examples | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | Min and Max ramping speed of a thermal generator. Minimum up time and down time of generation units. Up/down spinning reserve. | Optimization (EIPSO) | | to test the new algorithm. | |
| (Hetzer et al 2008), | Yes | Nil | Yes | Fuel cost minimisation | Nil | Real power balance constraints. | Thermal generator limits. Wind turbine limits. | Not Specified | MATLAB | Two thermal and Wind turbine generators | No |
| (Lu et al., 2008) | Yes | Nil | Yes | Fuel cost with valve point effect | Nil | Real power balance constraints. Conventional Unit's maximum up/down reserve contribution constraints. Thermal Unit's up/down spinning reserve contribution constraints. | System up/down spinning reserves. System Min/max power output constraints. Thermal plant capacity constraints. Conventional plant ramp-rate limits constraints. Wind power curve constraints. Total actual wind generation limit. Wind power penetration limit. Wind generation fluctuation constraints | Evolutionary Simulated Annealing (ESA) algorithm combined with the Direct Search Method (DSM) | FORTRAN | Not Mentioned | No |
| (Morales, et al 2009) | Yes | Nil | Nil | Minimize fuel cost | Nil | Real power balance constraints. | Constraints Pertaining to the Electricity Market and Involving First-Stage Variables. Constraints pertaining to the actual system operation and involving second-stage variables. Constraints linking the market and the actual system operation, **involving first and second-stage variables.** | Stochastic Programming(SP) | Not Mentioned | Not Mentioned | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| (Kuo, 2010) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission loss constraints. | Ramp rate limits. Prohibited operating zone and generation limits. | Strength Pareto Evolutionary Algorithm (SPEA) | MATLAB | TAI-POWER and IEEE 30-bus systems | No |
| (Liu and Xu 2010a) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Total actual wind generation limit. Thermal plant capacity constraints. | Incomplete gamma(Probability Approach) | MATLAB | Standard IEEE 30-bus test system. Case studies included six thermal generators and two wind turbines. | No |
| (Liu and Xu 2010b) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Total actual wind generation limit. Thermal plant capacity constraints. | Incomplete gamma(Probability Approach) and Lagrangian Approach | MATLAB | Case studies included six thermal generators and one wind turbine unit. | No |
| (Liu., 2010) | Yes | Nil | Yes | Minimize fuel cost | Nil | Real power balance constraints. | Total actual wind generation limit. Thermal plant capacity constraints. | Incomplete gamma(Probability Approach) Lagrangian Method | MATLAB | A case study considered has six thermal generators and one wind turbine. | No |
| (Yongping et al., 2010) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Balance supply and demand constraints. | The upper and lower limits of the generating units. | Singular Weighted Method (SWM) | MATLAB | Power system with five power-generating units. | No |
| (Jiang et al, 2011) | Yes | Nil | Yes | Fuel cost with valve point loading effect | Nil | Real power balance constraints. Transmission loss constraints | Real power operating limits of the generation unit. Spinning reserve constraint(USR and DSR) Generating unit ramp rate limits. | PSO | Not Specified | 6-unit system and the 15-unit system. | No |
| (Liu, 2011) | Yes | Nil | Yes | Minimize fuel cost | Nil | Real power balance constraints. | Real power operating limits of generation units. | Beta distribution of the PDF(Probabili | MATLAB | Not Mentioned | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | stic Approach) | | | |
| (Li & Jiang. 2011) | Yes | Nil | Yes | Minimize fuel cost | Nil | Real power balance constraints. Transmission constraints: | Generator limits constraints. Ramping rate constraints. Minimum up/down-time: | PSO | Not Mentioned | IEEE 30-bus power system. The Shanghai network which consists of a wind turbine and five thermal generating units | Yes |
| (Alhasawi, & Milanovic, 2012) | Yes | Nil | Yes | Minimize fuel cost | Nil | Power balance constraints. | Minimum and Maximum active and reactive power limits. Min and Max voltage magnitude. | Genetic Algorithm and Lagrangian Method. | MATLAB | Not Specified | No. |
| (Azizipanah-Abarghooee, et al 2012) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve loading effect and emission function | Power balance constraints. Transmission loss constraints | Real power operating limits of generation units. Prohibited operating zone. | Modified Teaching-Learning algorithm(MTLA) | Not Specified | (i.) The first system consists of 6 units, neglecting power losses. (ii.) The second system includes 14 units and the power losses are considered. (iii.) The third system comprises 40 units, of which 5 out of 40 exhibit prohibited zones, and the effect of power losses is neglected. | No |
| (Peng, et al 2012) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve effect and emission | Power balance and transmission loss constraints. | Prohibiting zones. Minimum and maximum operating limits of both wind and thermal generators. Ramp rate constraints. Minimum on/off time constraints. Ramping up/down rate constraints. | Bi-population Chaotic Differential Evolution (BPCDE) | Not specified | Power system with 5 thermal power units integrated with large-scale wind farms, which consist of 160 wind turbines. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| (Hafiz et al.,2012) | Yes | Nil | Yes | Minimize fuel cost | Nil | Power balance constraints. | Capacity constraints<br>Ramp rate constraints | Feasible Operation Region (FOR) | Not mentioned | Not specified | No |
| (Jadhav & Roy., 2013) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve point loading effect and emission function | Power balance and transmission loss constraints | Min/Max generators operating limits.<br>Prohibited operating zone of thermal generators.<br>Ramp rate limits of thermal generators | Gbest Guided Artificial Bee Colony algorithm (GABC) | MATLAB | IEEE 30 bus system. | No |
| (Mondal, et al., 2013) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve effect and emission | Power balance constraints | Max/Min operating limits of both wind and thermal generators. | Gravitational Search Algorithm (GSA) | MATLAB | IEEE 30-bus test system. | |
| (Reddy et al., 2013), | Yes | Nil | Yes | Minimize fuel cost with valve-point loading. | Nil | Power balance and transmission loss constraints | Total spinning reserve requirement constraints.<br>Max/Min operating limits of both wind and thermal generators.<br>Generator spinning reserve constraints. | Mean Learning Technique (MLT) | MATLAB | The algorithm is implemented on 6 units and 40-unit test systems. | No |
| (Zhao, et al 2013) | Yes | Hydropower( pumped storage) | Yes | Nil | Minimize fuel cost and emissions | Power balance constraints. | Generator spinning reserve.<br>Max/Min operating limits of wind and thermal generators.<br>Ramping rate constraints.<br>Minimum up/down time constraints. | General Algebraic Modelling System Branch-And-Reduce Optimization Navigator (G AMS BARON) solver Algorithm | General Algebraic Modelling System (GAMS) Software | Not Mentioned | No |
| (Li et al., 2013) | Yes | Nil | Yes | Minimize fuel cost | Nil | Power balance constraints. | Ramping rate constraint.<br>Generation output limit of thermal units.<br>The generation output limit of wind power generators.<br>Positive spinning reserve constraint. | Lagrangian Approach | VC++ 9,0 | Not Mentioned | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | Up/Down spinning reserve constraints. | | | | |
| (Niu and Wei, 2013) | Yes | Nil | Yes | Nil | Minimize fuel cost and emission | Power balance constraints. Transmission losses | Power operating limits. Spinning reserves. | Karush-Kuhn-Tucker(KKT) and Hybrid Particle Swarm Optimization( HPSO) | Not specified | Not specified | No |
| (Zhu, et al .,2014) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Power balance constraints. | Reserve capacity constraint. Units output constraint for both thermal and wind generators | Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D) | MATLAB | 6-generator system and a 40-generator system with wind farms. | No |
| (Dubey, et al 2015) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve point loading effect and emission function | Power balance constraints. Transmission losses | Ramping rate constraint. Generation output limit of thermal units and wind turbines. | Flower Pollination Hybrid Algorithm (HFPA)-TVFSM( Time Varying Fuzzy Selection Mechanism) | Not specified | Tested on two wind-thermal power systems | No |
| (Han, et al., 2015) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve point loading effect and emission function | Power balance constraints. | Lower and upper generating capacities for thermal and Wind generator. Ramp rate limits of thermal generators. | Diffusion Particle Optimization (DPO) | MATLAB | (i). Thirteen thermal generating system without considering emissions and wind power. (ii)Test system with a modified IEEE 30-bus/6-generators test system. (iii) The test system consists of an economic dispatch problem considering emissions. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| (Jiang, et al., 2015) | Yes | Nil | Yes | Nil | Minimize fuel cost with valve point loading effect and emission function | Power balance constraints. Transmission losses | Lower and upper generating capacities of the thermal and wind generator system. Ramp rate limits of thermal generators. Prohibited operating zone of thermal generators. | Gravitational Acceleration Enhanced Particle Swarm Optimization algorithm (GAEPSO). | MATLAB | 40 thermal generators and 2 wind turbines test system. | No |
| (Liang, et al., 2015) | Yes | Nil | Yes | Minimize fuel cost. | Nil | Power balance constraints. Transmission losses | Minimum/Maximum thermal generator reactive and active power output. Min/Max wind turbine output. Min/max voltage magnitude output. Min/max voltage phase angle. Minimum/Maximum load tap changer position of the power transformer. Minimum/Maximum reactive power injection using capacitor banks. The maximum line flow of the transmission line. | Firefly Algorithm (FA) | MATLAB | The IEEE 30-bus system. | No |
| (Meyyappan and Pandu, 2015) | Yes | Nil | Yes. | Minimize operational cost. | Nil | Power balance constraints. | Lower and upper generating capacities for thermal and Wind generator system. | Primal Dual Interior Point(PDIP), Differential Evolution(DE) , and Bacterial Foraging Technology (BFT). Using ANN and wavelet transform to predict the wind forecast(WNN | MATLAB | Three, 13, and 40 thermal units systems, along with three wind generators of the same rating. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | )( Wavelet neural network) | | | |
| (Pandit, et al 2015) | Yes | Nil | Yes | Nil | Minimize operating cost with valve point loading effect and emission function | Power balance constraints. Transmission losses. | Lower and upper generating capacities for the thermal and wind generator system. Ramp rate limits of thermal generators. Prohibited operating zone of thermal generators | PSO-DE | MATLAB | 30 bus system with six conventional thermal generating units and one wind farm which consists of 20 similar wind turbines | No |
| (Geetha, et al., 2015) | Yes | Nil | Yes | Minimize operating cost. | Nil | Power balance constraints. | Lower and upper generating capacities for thermal and Wind generator systems. Ramp rate limits of thermal generators. Minimum on/off time constraints. | Genetic Algorithm. | MATLAB Fuzzy logic model. | Genco system comprises ten thermal units and two wind farms. | No |
| (Güçyetmez & Çam, 2016) | Yes | Nil | Yes | Minimize operating cost. | Nil | Power balance constraints. Transmission losses | Minimum/Maximum thermal generator reactive and active power outputs. Min/Max wind turbine output. Min/max voltage magnitude. Min/max voltage phase angle. | Hybrid Genetic-Teaching Learning-Based Optimization (G-TLBO) algorithm | MATLAB/ SIMULIN K | Thermal-wind-powered Turkish energy generation system of 19 buses | No |
| ( Alham et al, 2016) | Yes | Yes | Yes | Nil | Minimize operating cost and | Power balance constraints. Transmission losses. | Lower and upper generating capacities for thermal and Wind generator. | Genetic Algorithm (GA) | The GAMS software/ MATLAB | Standard IEEE 30-bus system. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | emissions . Several cases as the fuel cost minimization, Emission minimization, and emission and cost minimization simultaneously | Load demand without /with using DSM. | Ramp rate limits of thermal generators. Constraints of energy storage. Upward demand variation and downward demand variation at each hour of DSM(Demand Site Management) | | | | |
| (Ghasemi, et al., 2016) | Yes | Nil | Yes | Nil | Minimise Fuel and emission cost | Real power balance constraints. Transmission losses. | Minimum/Maximum thermal generator reactive and active output. Min/Max wind turbine output. Min/max voltage magnitude output. Prohibited operating zone of thermal generators Transmission line security constraints. | Online Honey Bee Matting optimisation (HBMO) (OLHBMO) | MATLAB | IEEE 30-bus 6-unit, the IEEE 118-bus 14-unit, and the 40-unit with valve point loading effect test systems | No |
| (Krishnasamy & Nanjundappan, 2016) | Yes | Nil | Yes | Minimize fuel cost with valve-point loading. Multiple fuel option is considered | Nil | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator system. | Weighted Probabilistic Neural Network (WPNN) - Biogeography -Based Optimization (BBO) algorithm | MATLAB | Not Mentioned | Yes |
| (Qu et al, 2016) | Yes | Nil | Yes | Minimize fuel cost with | Nil | Real power balance constraints | Lower and upper generating | Summation-based Multi-Objective Differential | MATLAB | (i) IEEE 30-bus 6-generator system. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | capacities of thermal and wind generator systems. | Evolution (SMODE) algorithm | | (ii) IEEE 30-bus 6-generator system including the transmission losses. (iii) 40-thermal-generator lossless system. | |
| (Velamuri et al, 2016) | Yes | Nil | Yes | Minimize fuel cost. | Nil | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. | Flower Pollination Algorithm (FPA) | MATLAB | IEEE 30 bus system. | No |
| (Chen et al.,2017) | Yes | Nil | Yes | Nil | Minimise Fuel and emission cost | Real power balance | Lower and upper generating capacities of thermal wind, and hydro generating systems | Modified Gravitational Search Algorithm based on the Non-dominated Sorting Genetic Algorithm-III (MGSA-NSGA-III) | MATLAB | Four cascaded hydropower plants, three thermal plants, and two wind power plants. | No |
| (Hu et al.,2019) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints | Lower and upper generating capacities of thermal and wind generator systems. Ramp rate limits of thermal generators. | Genetic-Sequential Qaudratic Programming Algorithm(GA-SQP) | MATLAB | IEEE 30 bus test system with six thermal generators and large scale wind-farm | No. |
| (Jiang et al., 2019) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. | Gravitational-Particle Swarm Optimisation Algorithm (GPSOA) | MATLAB | IEEE 30 bus test system with six thermal generators and two wind turbines | No |
| (Behera et al.,2020) | Yes | Nil | Diesel and Solar | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Lower and upper generating | Constriction Factor-Based Particle Swarm | MATLAB | Six bus systems with three TGs, two | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | capacities of thermal, Solar, and wind generating systems | Optimization (CFBPSO) | | WPGs, and two SPGs | |
| (Jin et al., 2021) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. | Technique for Order Preference by Similarity to an Ideal Solution(TOPSIS) | MATLAB | Six classical generating units and one wind system are used. | No |
| (Salkuti., 2021) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. Ramp rate limits of prohibited zones for thermal generators. | Prticle Swarm Optimisation( PSO) | MATLAB | Six thermal generating units, but later generator one is replaced by a wind power system | No |
| (Yan et al., 2022) | Yes | Yes Electrical Vehicle (EV) | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. Ramp rate limit, spinning reserves for thermal generators. Travel constraints of EV owners, EVs' remaining power constraints, and charging and discharging EVs. | Self-Adaptive Multi-Learning Harmony-Search Algorithm (SAMLHS) | MATLAB | Ten thermal units were used, and a Toyota RAV4 served as the EV. Different EV-to-Wind scenarios are used. | No |
| (Nagarajan et al.,2022) | Yes | Nil | Diesel and Solar | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Lower and upper generating capacities of thermal, Solar, and wind generating systems | Improved Mayfly (IM) Algorithm | MATLAB | Three conventional generators, solar, and wind units | No |
| (Azeem et al.,2023) | Yes | Nil | Diesel and Solar | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Lower and upper generating capacities of thermal, Solar, and wind generating systems | Chaotic Salp Swarm Algorithm (CISSA) | MATLAB | Different test cases, scenarios, and combinations of thermal, solar, and wind units. | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Zou et al., 2023). | Yes | Nil | Diesel and Solar | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Lower and upper generating capacities of thermal, Solar, and wind generating systems. Ramp rate limit, **spinning reserves for** thermal generators. Heat-generating limits, capacity limits of the CHP units | Global Nondominated Sorting Genetic Algorithm II (GNSGA-II) | MATLAB | Four CHPDEED cases with or without renewable energies are considered. | No |
| (Lalhmachhuana et al.,2024) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints | Lower and upper generating capacities of thermal and wind-generating systems. Security constraints, minimum and maximum reactive power | Multi-Objective Particle Swarm optimisation( MOPSO) and Multi-Objective Ant-Lion optimisation( MOALO) | MATLAB | IEEE-30 bus system, Ten 3.2MW wind power system | No |
| (Secui et al., 2024) | Yes | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind-generating systems. | Modified Social Group Optimization (MSGO) | MATLAB | IEEE30 bus system with 10 and 40 units. The wind power consists of two wind systems, each having 550MW | No |
| (Qiao et al., 2025) | Yes | Yes Electrical Vehicle (EV) | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. Transmission losses. | Lower and upper generating capacities of thermal and wind generator systems. Ramp rate limit, spinning reserves for thermal generators. Travel constraints of EV owners, EVs' remaining power constraints, and charging and discharging EVs. | Self-adaptive Multi-mode Teaching-Learning-Based Optimization (SaMmTLBO) algorithm | MATLAB | Ten thermal units were used, and a Toyota RAV4 served as the EV. Different EV-to-Wind scenarios are used. | No |
| (Xiong et al.,2025) | Yes | Nil | Diesel , | Nil | Minimize fuel cost | Real power balance constraints. | Lower and upper generating | Multi-Objective | MATLAB | Modified IEEE 30-bus system with two thermal | No |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Solar, and hydro power | | and emissions | Transmission losses. | capacities of thermal wind, hydro, and solar generating systems, constraints, and minimum and maximum reactive power constraints. | Artificial Gorilla Troops (MOAGT) optimizer | | units. One wind and SRH power plants The combined wind and SRH power plant and the Modified IEEE 118-bus system have 54 thermal units, three wind plants, and two solar plants. Combination of PV and SRH, and of Wind and SRH. | |
| (Meng et al., 2025) | Yes | Yes | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints. | Lower and upper fan power, SOC, wind, Photovoltaic, and transfer load. | Demand Response(DR) | Not mentioned | conventional, wind-only, wind+storage, wind+DR, and fully integrated. | No |
| (Li et al., 2024) | Yes | Yes | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints, which include: Electric Power Balance Constraints, Thermal Power Balance, and Gas Power Balance Constraints | Lower and upper, Photovoltaic, wind, Energy Storage Unit Operational Constraints, Purchased Energy Power Constraints, Hydrogen Equilibrium Constraints | Information Gap Decision Theory(IGDT) | Not mentioned | Six different scenarios are used to validate the effectiveness of the algorithm | No |
| (Mao et al., 2026) | Yes | Yes | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints, which include: | Lower and upper, Photovoltaic, wind, Energy Storage Unit Operational Constraints, Purchased Energy Power Constraints | Distributed Robust Optimization (DRO)-Model Predictive Control (MPC). | MATLAB | Uncertainty scenarios are consist of the PV, WT and fluctuations of diverse form of energy demands. | Real-time formula presented |

## 2.3 Hybrid Renewable Energy Systems (HRES) economic dispatch

The ever-rising oil prices, declining fuel resources, insatiable demand for electrical energy, and the worldwide concern for atmospheric pollution and environmental protection have caught the attention of many researchers. The use of Distributed Energy Resources (DERs) has caught the attention of many researchers. These resources are considered environmentally friendly and have lower operational cost. Power generation from such resources is consistent and could reduce the overall cost of a power system if they are operated and scheduled wisely. However, integrating these resources into the current thermal generation may disrupt some security constraints of the Economic Dispatch (ED) model of the existing system, due to the erratic and random nature of these DERs. Therefore, a correctly modelled system and more secure constraints need to be added to the thermal ED formulation to avoid casualties during power system operation.

A hybrid system's main goal is to provide users with dependable AC/DC power at a reasonable price. It can be set up as a grid-connected system or as an off-grid, standalone, or island HRES. Figure 2.11 displays many classes. According to (Theo et al., 2017)HRES is composed of various power-generating technologies, including solar PV, wind turbines, fuel cells, traditional diesel generators, biogas generation, biomass generation, and CHP.

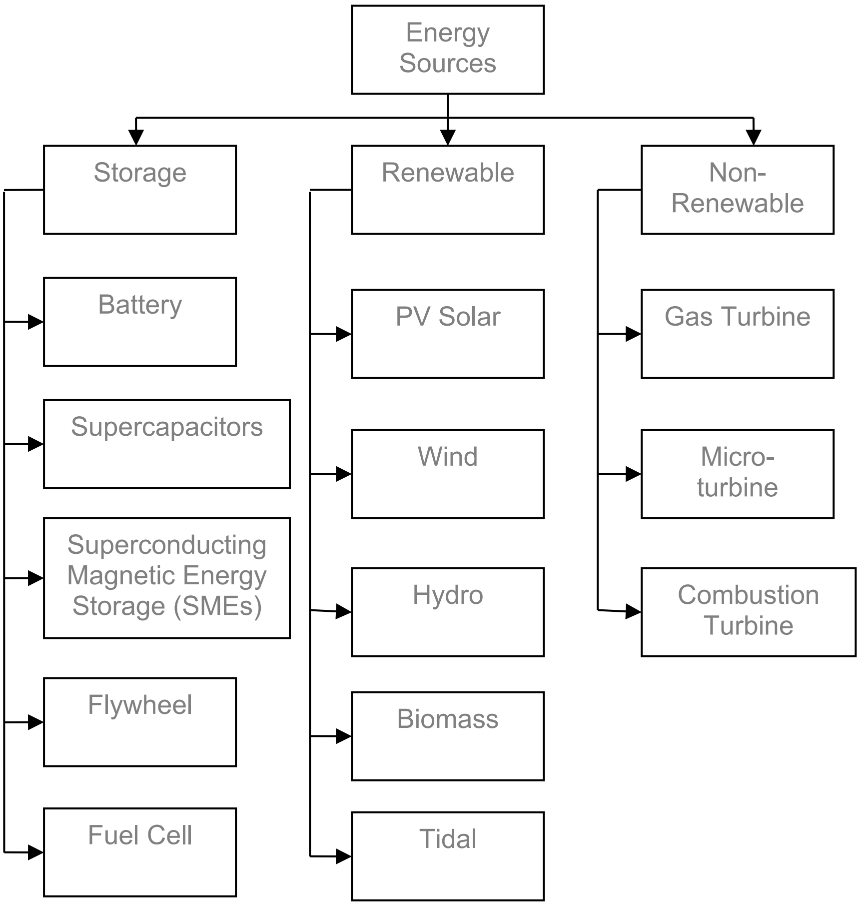



**Figure 2.11:** **Power sources and storage for a mixed renewable energy system** (Ogunjuyigbe et al., 2016)

A storage battery system is what makes an HRES better than a non-HRES. According to various authors, for example, Kaldellis (2010) states that the storage battery can be a capacitor, flywheel system, pumped hydro, or compressed air.

### 2.3.1 Study of techniques applied in solving the single-area economic dispatch problem for Hybrid Renewable Energy Systems (HRES)

#### 2.3.1.1 Single-criteria

The economic dispatch dependability of hybrid power systems has been examined using a variety of mathematical models and techniques with several goals, including cost minimization, pollution reduction, and dynamic load control.(Diaf et al., 2008), studied the optimum formation of a stand-alone hybrid photovoltaic-wind system that ensures the energy autonomy of a typical remote consumer with the lowest levelized cost of energy. The techno-economic optimization method is used to investigate different combinations. (Kashefi Kaviani et al., 2009), used a PSO to improve a hybrid wind-photovoltaic-fuel cell generation system to minimise the yearly cost of the hybrid system, subject to a reliable supply to meet the demand. (Giannakoudis et al., 2010) used an optimisation method based on heuristic algorithms called Simulated

Annealing(SA) for the design and operation of a hybrid power generation system that consists of wind generators, a diesel generator, PV panels, accumulators, an electrolysis apparatus, hydrogen storage tanks, a compressor, and a fuel cell.

(Kaabeche et al., 2011), suggested an optimisation model based on an iterative technique to optimise the size of a hybrid wind/photovoltaic system combined with a battery bank, to reduce power supply insufficiency and the levelized unit electricity cost, subject to system equality and inequality constraints. A two-stage adaptive robust optimization model for the Security Constrained Generation Scheduling (SCGS) problem, where the first-stage unit commitment decision and the second-stage ED decision are solved (ElDesouky, 2014).

An adaptive hybrid technique involving an Artificial Neural Network (ANN) and a Genetic Algorithm (GA) is used. The authors (Arabali et al., 2014) Used a stochastic optimization method to optimally size the hybrid system components while satisfying the system reliability requirements. A sequential Monte Carlo simulation (SMCS) method is used to create data in a time sequence and then analyse reliability indices from the simulated results. The SMCS method examines the system's sequential performance by generating consecutive samples of system states over multiple time intervals. This offers a better illustration of the hybrid system’s behaviour for reliability analysis. The Direct Search (DS) method, Pattern Search (PS), is used to solve the non-smooth optimal sizing problem for the hybrid power system.

The authors (Som & Chakraborty, 2014) Presented a power system that consists of fuel cells, solar modules, and biomass gasifier units as DERs. A facility for buying electricity in a contract with a utility is also considered in both types of power-delivery systems. The mathematical investigations are performed using a Real-Valued Cultural Algorithm (RVCA). The authors (Arriagada et al., 2015) Presented a probabilistic economic dispatch model considering thermal units (fuel generators), photovoltaic arrays, and wind energy conversion systems. The thermal generator is modelled with a quadratic cost function. The Weibull distribution is used to represent the wind power output. Lastly, the PV plant is modelled with a Beta distribution.

In this study, the authors (Azizipanah-Abarghooee et al., 2015), a CCP (chance constrained programming) based on the (Jointly Distributed Random Variables Method (JDRVM) is considered to solve the problem of stochastic Multi-Objective Combined Heat and Power Economic Dispatch (SMCHPED), including random behaviours of wind energy conversion system and photovoltaic output power, as well as electrical and thermal load demand. The uncertainty of each random variable is modelled by a PDF (Probability Density Function). The output power risk of WPGs and PVUs can be covered by other thermal units and Combined Heat and Power (CHP) devices.

The authors (Plathottam & Salehfar, 2015) Formulated an economic dispatch problem for a power system control area with more diffusion of renewable production and power storage. The PV/wind/storage system considered has 20% wind generation and 10% distributed PV compared to the overall generation capacity. The cost functions and constraints related to each component are considered. Simulated Annealing(SA) and PSO are used to solve the optimization problem.

The authors (Malheiro et al., 2015), Formulated economic dispatch(ED) to minimise the operation cost of the power system subject to inequality and quality constraints, using the Mixed-Integer Linear Programming technique. A battery bank, wind, PV solar, and a backup diesel generator make up the HRES.

In the study of (Maleki & Askarzadeh, 2014)The discrete harmony search approach was used to solve the ED problem with the wind/PV solar/diesel/battery system. The simulation results were compared using the Discrete Simulated Annealing (DSA) methodology. (Belmili et al., 2014), examined sizing techniques for independent photovoltaic-wind hybrid systems. The first section of the article presents various software for addressing the HRES issue (HYBRID2, HYBRIDS, HOMER (Hybrid Optimisation Model for Electric Renewable), HOGA (Hybrid Optimisation by Genetic Algorithms. It provides modeling of various HRES components and solutions in the context of power supply balance constraints, including inequality and equality constraints. To approximate all aspects of electricity consumption, the TRNSYS software is used to simulate the dynamic thermal behaviour of residential buildings and forecast the electrical energy required to meet the thermal load, which partly contributes to the non-controllable load of the residential area.

(Atia & Yamada, 2016). The probability distributions of electric load and renewable energy resources are described by Probability Distribution Functions (PDFs) and introduced into the optimization using scenario generation techniques. The optimisation method uses a mixed-integer linear programming (MILP) approach. The authors (Cao et al., 2016)presented a weather-dependent optimal power flow algorithm with wind farm integration that considers temperature-related resistance and Dynamic Line Rating (DLR) of overhead lines.

The authors (Guo et al., 2014) studied a stand-alone microgrid with a seawater desalination system. An EMS for a stand-alone micro-grid consisting of a WT generator, a diesel generator, an ESS, and a seawater desalination system. The energy management system (EMS) can control supply-demand balance and maximize the environmental or economic benefits. The stand-alone wind-diesel micro-grid uses an EMS strategy that optimizes the charging/discharging cycles of the storage system and the system's operational cost based on predictions of wind turbine output and load demand. Neural Network-based wind forecasting is used to forecast real-time wind data, and the Genetic Algorithm is used as an optimization method.

(Theo et al., 2017) and (Kusakana, 2019), provides a thorough analysis of various hybrid system configurations, examining modelling, reliability, optimisation techniques, and optimal operation control.

#### 2.3.1.2 Multi-criteria

(B. Lu & Shahidehpour, 2005), created a short-term grid-connected PV/battery system using the Lagrangian method through a security-constrained unit commitment algorithm. The terminal current-voltage (I-V) characteristics of solar cells are established through a two-diode model and equality/inequality constraints. In (Liang & Liao, 2007)The optimal control problem is defined with respect to PV solar, wind, conventional, and hydropower, with environmental pollutants such as and . Power balance is taken into account as inequality constraints. The problem is solved using a fuzzy-logic optimization approach.

The authors (Dufo-López & Bernal-Agustín, 2008; Shayeghi et al., 2015) also present papers based on a multi-objective optimization problem, including cost minimization and carbon dioxide emissions, and the various mathematical models used are derived from previous research.

(Dufo-López et al., 2007; Dufo-López & Bernal-Agustín, 2005, 2008) present the HRES problem as a synthesis of a PV-Wind-Diesel-Hydrogen-Battery system, where a Multi-Objective Evolutionary Algorithm (MOEA) and a Genetic Algorithm (GA) are used to find the optimal solution for hybrid system synthesis and control.

The authors (Shayeghi et al., 2015) Used the Artificial Bee Colony(ABC) method. In (Ogunjuyigbe et al., 2016)The objective is to minimise life-cycle cost, energy losses, and carbon dioxide emissions, with the GA used to find solutions subject to component constraints. The total power generation from PV, wind, Diesel, and stored energy must always be greater than the load demand. A three-split diesel generator model is used to replace standard single diesel units, enabling multiple generators to meet the user's demands at specified times while minimising use at low load factors.

(Kamjoo et al., 2016), solved the stand-alone HRES optimisation problem, which synthesises solar, wind, and battery bank using Non-dominated Sorting Genetic Algorithm (NSGA-II) in combination with Chance Constrained Programming (CCP). The nonlinear constraints require that the summed output powers from both PV and wind must provide the overall load balance, or otherwise, the stored power in the battery must. The nonlinear constraints depend solely on the battery bank's state of charge and the physical constraints of the other two components of the HRES.

According to the authors' research (Abul'Wafa, 2013), the dispatch problem includes wind/ PV solar/ thermal units, as well as emissions. The wind and solar output power are presented using the Weibull distribution. The underestimation and overestimation of PV and wind are included in the objective cost function. The conventional generators and emissions are represented by a quadratic cost function with valve-point loading and by a quadratic function with an exponential term, subject to equality and inequality constraints.

(Bilil et al., 2014), proposed the probabilistic economic dispatch (EED) problem for a power system integrating RESs. The main objective is to minimize the cost of power generation and the greenhouse gas emissions of the system, subject to the operation limits of RESs. The multi-objective optimisation problem is investigated using CE-NSGA-II. The review summary of the HRES economic dispatch is given in Table 2.4. Figure 2.12 illustrates the graphical representation of Table 2.4. This figure shows the number of publications per year.

**Table 2.4:** **Quantity of paper for HRES economic dispatch on a yearly basis**

| Quantity of papers year-wise | | |
|---|---|---|
| **Reference** | **Year of publication** | **Number of publications** |
| (Lu & Shahidehpour,2005), (Dufo-López, & Bernal-Agustín, 2005) | 2005 | 2 |
| (Dufo-López, & Bernal-Agustín, 2006) | 2006 | 1 |
| (Liang, and Liao., 2007), (Dufo-López et al., 2007) | 2007 | 2 |
| (Diaf et al., 2008), (Dufo-Lopez, & Bernal-Agustin.,2008) | 2008 | 2 |
| (Kashefi Kaviani et al., 2009) | 2009 | 1 |
| (Giannakoudis et al., 2010), (Kaldellis.,2010) | 2010 | 2 |
| (Kaabeche et al., 2011) | 2011 | 1 |
| (Shengwei Mei et al., 2012) | 2012 | 1 |
| ( Abul'Wafa, 2013) | 2013 | 1 |
| (Som & Chakraborty., 2014), Trifkovic., et 2014), (Bilil et al, 2014) (ElDesouky, 2014), Arabali et al., 2014), Belmili et al., 2014), (Maleki & Askarzadeh., 2014) | 2014 | 7 |
| (Arriagada et al., 2015), (Plathottam, & Salehfar., 2015) (Azizipanah-Abarghooee et al., 2015), (Arriagada et al., 2015), (Malheiro et al.,2015) | 2015 | 5 |
| ((Kamjoo et al., 2016), (Guo et al., 2016), (Cao et al., 2016), (Atia & Yamada.,2016), (Ogunjuyigbe et al.,2016), (Kusakana., 2016) | 2016 | 6 |

| | | |
|---|---|---|
| (Theo., 2017) | 2017 | 1 |

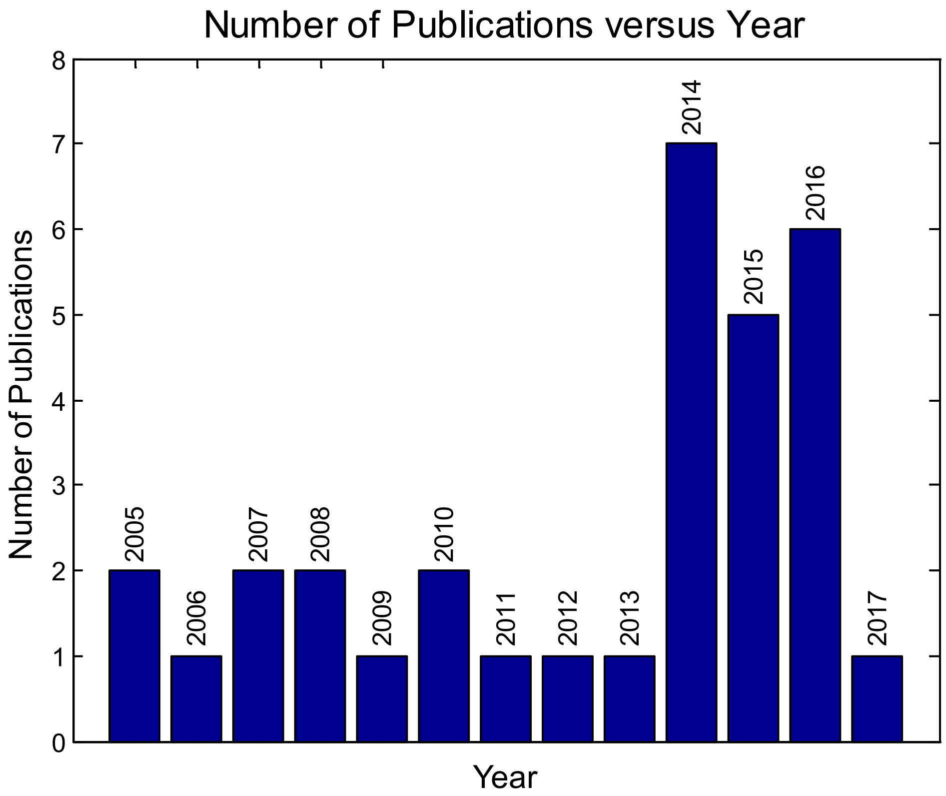



**Figure 2.12:** **Number of publications of HRES economic dispatch on a yearly basis**

**Table 2.5:** **Techniques and algorithms applied to the HRES single-area economic dispatch problem.**

| **Most used methods and algorithms** | | **Reference paper** | **No. of publications** |
|---|---|---|---|
| **Algorithm** | **Abbreviation** | | |
| Artificial Neural Network | ANN | (ElDesouky, 2014), (Guo et al., 2016) | 2 |
| Controlled Elitist – Non-Dominated Sorting Geneti Algorithm-II | CE-NSGA-II | ( Abul'Wafa, 2013), (Bilil et al, 2014), (Kamjoo et al., 2016) | 3 |
| Differential Evolution | DE | (Basu et al., 2012) | 1 |
| Fuzzy optimization method | FL | (Liang, and Liao., 2007) | 1 |
| Game theory | GT | (Mei,et al., 2012) | 1 |
| Genetic algorithm | GA | (ElDesouky, 2014), (Guo et al., 2016), (Ogunjuyigbe et al.,2016), (Dufo-Lopez, & Bernal-Agustin.,2008) | 4 |
| Harmony Search | HS | (Maleki & Askarzadeh., 2014) | 1 |
| Hybrid Modified Cuckoo Search Algorithm And Differential Evolution | H-MCSA-DE | (Azizipanah-Abarghooee et al., 2015) | 1 |
| Iterative Optimisation | IO | (Kaabeche et al., 2011) | 1 |
| Iterative Optimization Technique | IOT | (Kaabeche et al., 2011) | 1 |

| | | | |
|---|---|---|---|
| Lagrangian Method | LR | (Lu & Shahidehpour,2005) | 1 |
| Mix Integer Linear Programming | MILP | (Malheiro et al.,2015) | 1 |
| Mixed Integer linear Programming | MIP | (Atia & Yamada.,2016) | 1 |
| Multi Objective Evolutionary Algorithm | MOEA | (Dufo-Lopez, & Bernal-Agustin.,2008) | 1 |
| Particle swarm optimisation | PSO | (Kaviani et al., 2009), (Plathottam, & Salehfar., 2015) | 2 |
| Pattern Search-based optimization method | PS | Arabali et al., 2014) | 1 |
| Probalistic Method | PM | (Arriagada et al., 2015) | 1 |
| Real Valued Cultural Algorithm | RVCA | (Som & Chakraborty., 2014) | 1 |
| Sequential Monte Carlo simulation | SMCS | Arabali et al., 2014) | 1 |
| Simulate Annealing | SA | (Giannakoudis et al., 2010), (Plathottam, & Salehfar., 2015), (Maleki & Askarzadeh., 2014) | 3 |
| Techno-Economical optimisation | TE | (Diaf et al., 2008), (Belmili et al., 2014) | 2 |
| Weather-Based Optimal Power Flow algorithm | WB-OPF | (Cao et al., 2016) | 1 |

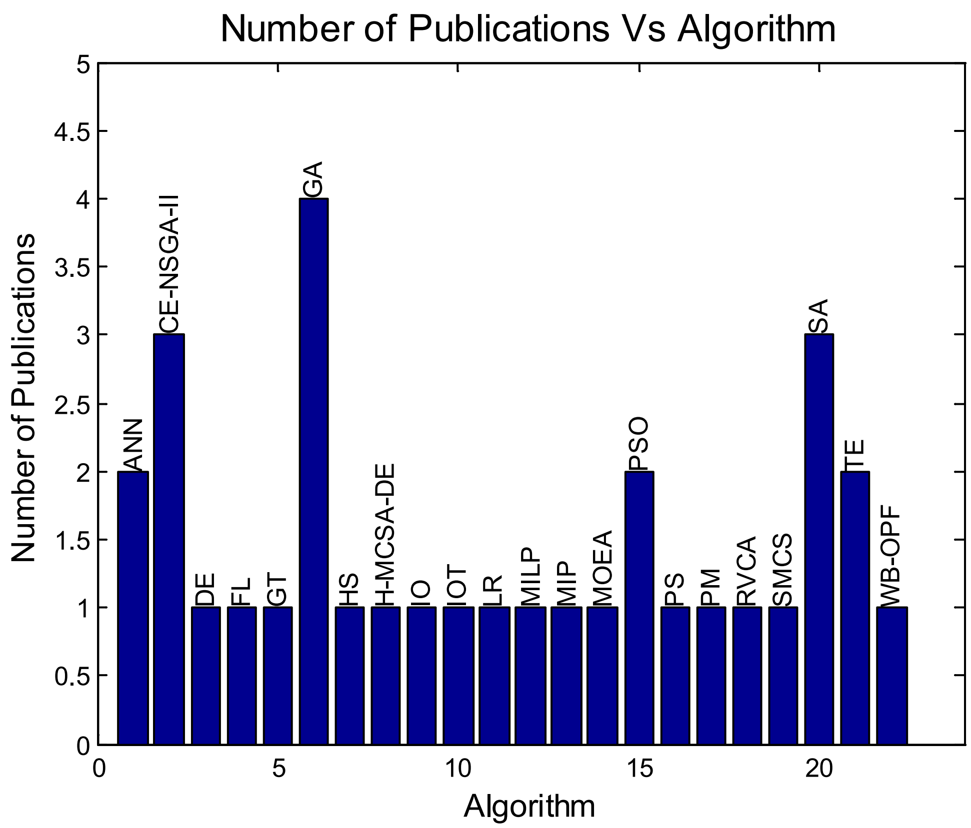



**Figure 2.13:** **The quantity of papers pertaining to the HRES economic dispatch algorithms**

Table 2.6 provides a summary of the review for single-area, single-criteria, and multi-criteria hybrid renewable systems.

**Table 2.6:** **Study of the literature for single and multi-criteria mix renewable energy systems.**

| Reference Paper | System Component | | | | | | Objective Function | | Constraints | | Algorithm | Software | Power System Considered | Real life implementation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PV Panel | Wind | Fuel cell/ other | Hydro Power | Storage | Diesel & Other | Single objective | Multi objective | Equality Constraints | Inequality Constraints | | | | |
| (Lu & Shahidehpour,2005) | Yes | Nil | Nil | Nil | Yes | Yes | Minimize operation cost | Nil | State of charge power balance and power balance constraints. | State of charge limits. Charge/discharge current limits. Output power limits | Lagrangian | Not mentioned | Eight-bus system. | **No** |
| (Liang and Liao., 2007) | Yes | Yes | Nil | Yes | Yes | Yes | Nil | Minimize operation cost and emissions | Power balance constraints. | Spinning reserve constraints for hydro and thermal units. Capacity limits for thermal generator with and without prohibiting zones Min/max of water volume released from each reservoir. Power output limits of the wind energy system. Power output limits on the solar energy system. | Fuzzy optimization method | MATLAB | Not specified | No |
| (Diaf et al., 2008) | Yes | Yes | Nil | Nil | Yes | Yes | Optimum size of the system, able to fulfil the energy requirements of a given load distribution | Nil | Power balance equations constraints. | Maximum and Minimum allowable storage capacity constraints. Min/Max constraints of PV and wind generator systems | Techno-economic optimisation method | Not specified | Three selected sites located in Corsica Island (Calvi, Ajaccio, and Cape Corse) | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Dufo-Lopez, & Bernal-Agustin., 2008) | Yes | Yes | Nil | Nil | Yes | Yes | Nil | Minimise cost and emission, and unmet load | Power balance constraints. | Maximum and Minimum allowable storage capacity constraints. Min/Max constraints of PV and wind generator systems | Multi-Objective Evolutionary Algorithm (MOEA) and a Genetic Algorithm (GA) | Not Specified | PV-Wind-Diesel-Hydrogen-Battery system in Zaragoza | No |
| (Kaviani et al., 2009) | Yes | Yes | Nil | Nil | Yes | Yes | Minimize operation cost. | Nil | Power balance constraints. | Minimum and Maximum capacity of PV, WTs, electrolyser, hydrogen tank, and Fuel cell. | PSO | MATLAB | System consisting of 14 wind generating units and 199 Photovoltaic arrays. | No |
| (Giannakoudis et al., 2010) | Yes | Yes | Nil | Nil | Yes | Yes | Minimize operation cost. Two different case studies were used, with changing design variables and uncertain parameters. The load was either constant or variable. | Nil | Power balance constraints. | Minimum and Maximum capacity of Diesel, PV, WTs, electrolyser, hydrogen tank, and Fuel cell. | SA | Grid computing environment | (i)Case I: 21 PV panels, 7 wind turbines, 15 Accumulators, one Fuel cell, and 0.3 Buffer tank. (ii) Case I: 27 PV panels, wind turbines 11, 6 Accumulators, one Fuel cell, and 0.2 Buffer tank. | No |
| (Kaabeche et al., 2011) | Yes | Yes | Nil | Nil | Yes | Nil | Minimize operation cost. | Nil | Nil | Minimum and Maximum capacity of PV, WTs. State of charge(SOC) minimum and maximum of the battery bank. | Iterative optimization technique | MATLAB | Arco-Solar with a max power rating of 43W, an AIR 403 wind turbine with 400W of rated power, and a Varta Solar battery with 100Ah capacity | No |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Mei,et al., 2012) | Yes | Yes | Nil | Nil | Yes | Nil | Nil | Minimise the total cost of generation and maximize the annual power selling income. To minimize the payment for power loss | Real power balance constraints | The battery's energy shouldn't be below the minimum permitted storage level. Also, it shouldn't exceed the batteries' planned capacity. | Game theory | Not specified | A fictitious hybrid power system consisting of wind turbines and PV panels. | No |
| (Basu et al., 2012) | Nil | Nil | Yes | Nil | Nil | Yes | Nil | Minimize fuel cost and emissions | Real power balance constraints | Bus Voltage Tolerance Limit. Limit on the active and reactive power Generation of the DERs. Line Flow Limits. | Differential Evolutionary | MATLAB | 14-bus radial micro-grid With 4 DERs. | No |
| (Abul'Wafa, 2013) | Yes | Yes | Nil | Nil | Nil | Yes | Nil | Minimize operating cost with valve-point loading and emission functions. | Real power balance constraints. | Minimum and maximum operational constraints for solar, wind, and conventional generating systems. | Controlled elitist NSGA-II (CE-NSGA-II) | MATLAB | Six Fossil-Fuel-Fired generators (FFGs), two Wind Energy Conversion Systems (WECS), and two photovoltaic (PV) systems are used. | No |
| (Bilil et al, 2014) | Yes | Yes | Nil | Nil | Nil | Yes | Nil | Minimize fuel cost and emission | Real power balance constraints. | Min/Max magnitude voltage and active power. | Controlled elitist NSGA-II (CE-NSGA-II) | Not restricted | IEEE 30-bus test system in two cases, with and without Renewable Energy Sources( RESs). | No |
| (ElDesouky, 2014) | Yes | Yes | Nil | Nil | Nil | Yes | Minimize operating cost. | Nil | Real power balance constraints. | Real power operating limits of thermal, wind and PV generating units. Unit minimum up/down (MUT/MDT) time for thermal generating units. | Artificial Neural Network (ANN)with Genetic Algorithm | MATLAB | Modified IEEE 24-bus system without hydro generation. Its transmission network consists of 24-bus location connected by 38 | No |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | The transmission line constraints. The ramp rate constraints for thermal generating units | (GA) and a priority list (PL), ANN/GA/PL. | | lines and transformers. | |
| Arabali et al., 2014) | Yes | Yes | Nil | Nil | Yes | Nil | Minimize operating and maintenance cost. | Nil | Real power balance constraints. | Storage Constraints. Min/Max operational constraints of Wind and PV units. | A pattern search-based optimization method is used in conjunction with a sequential Monte Carlo simulation (SMCS) | Not specified | Not specified | No |
| (Belmili et al., 2014) | Yes | Yes | Nil | Nil | Yes | Yes | Minimise cost of the system | Nil | Real power balance constraints. | Storage Constraints. Min/Max operational constraints of Wind and PV units | Techno-economic Algorithm | HOMER | Not specified | No |
| (Som & Chakraborty., 2014) | Yes | Yes | Yes | Nil | Yes | Nil | Minimize the total operating cost | Nil | Real power balance constraints | Upper and lower limits of DERs | Real Valued Cultural Algorithm (RVCA) | Not specified | Two different cases of DERs, i.e., case 1 Biomass Gasifier Unit (BMGU) and Solar Photovoltaic System (SPS) as DERs) and case 2 Phosphoric Acid Fuel Cells (PAFC )and SPS as DERs) was conducted. | No |
| (Maleki & Askarzadeh., 2014) | Yes | Yes | Yes | Nil | Yes | Nil | Minimise the total annual cost | Nil | Power balance constraints. | Upper and lower limits of DERs | Harmony Search(HS) and Simulated | MATLAB | System with PV, wind turbine and battery storage | No |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | Annealing(SA) | | | |
| (Shayeghi et al.,2015) | Yes | Yes | Nil | Nil | Yes | Nil | Nil | Minimise cost and emission | Power balance constraints | Upper and lower limits of DERs | Artificial Bee Colony (ABC) | MATLAB | System with PV, wind turbine and battery storage | No |
| (Arriagada et al., 2015) | Yes | Yes | Nil | Yes | Nil | Yes | Minimise fuel cost | Nil | Power balance constraints. | Min/Max constraints for wind, PV and thermal generating units. | Probabilistic Method | MATLAB | 17 thermal units , 205 MW of wind energy, 160 MW of solar PV and 5% and1% of thermal and NCRE failure rates, respectively. The wind turbine model used is 5 MW G132 (cut-in: 3 m/s, cut-off 25 m/s). The solar PV module used is the STP290-24 (290 W nominal power) and PV inverter used isSC 500HE (500 kW rated power). | Yes |
| (Azizipanah-Abarghooee et al., 2015) | Yes | Yes | Nil | Nil | Nil | Yes | Minimizing both the operation cost and risk level | Nil | Power balance constraints Heat balance constraint | Min/Max constraints for wind, PV, thermal, and CHP generating units. | Hybrid MCSA-DE (Modified Cuckoo Search Algorithm And Differential Evolution) | MATLAB | 6- and 40-unit test systems respectively. | No |
| | | | | | | | | | | Ramp rates, operating levels and | | | The power system consists of three different nodes. (i) Node A, consists of 2x500MW of | |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Plathottam, & Salehfar., 2015) | Yes | Yes | Nil | Yes | Yes | Yes | Minimize the operation cost | Nil | Power balance constraints Transmission losses. | min/max constraints of the conventional units. Min/max reservoir volumes of hydro plants. Min/Max operating constraints of DERs. Conventional Spinning reserves | Simulated Annealing(SA) and Particle Swarm Optimization(PSO) algorithms | MATLAB | thermal plant, Load (30%&20%) (ii) Node B consists of 4x200MW of Hydro plant and distributed solar PV of 200MW with 20% load. (iii) Node C, consists of 50MW of pumped hydro, 500MW of wind farm, load(20%) and tie line connecting neighboring control area | No |
| (Malheiro et al.,2015) | Yes | Yes | Nil | Nil | Yes | Yes | Cost Minimisation | Nil | Power balance constraints | Min/Max output power of wind, and PV. Min/max permissible storage of the battery. | Mixed Integer linear Programming(MIP | GAMS | The study consists of Diesel, PV, and wind generating units | No |
| (Atia & Yamada.,2016) | Yes | Yes | Nil | Nil | Yes | Yes | Reduce operation cost. | Nil | Power balance constraints. | Dispatched power by the maximum available generation for PV and WECS constraints. maximum contracted power constraints. Min/Max SOC for BESS | Mixed Integer linear Programming(MIP) | MATLAB | A residential microgrid in Okinawa with a daily demand of 4000 kWh. | No |
| (Cao et al., 2016) | Nil | Yes | Nil | Nil | Nil | Yes | Minimize the generation costs. | Nil | Power balance constraints Transmission losses. | Min/Max reactive and active power for thermal generators. Wind generator Min/max constraints. | Weather-Based Optimal Power Flow (WB-OPF) algorithm | MATLAB/ MATPOWER | Modified the IEEE 9-node system and the New England transmission Grid system | No |
| | | | | | | | Maximize the utilization of wind energy and minimize | Nil | | Min/Max generator limits. Min/Max SOC constraints and | Genetic Algorithm-BP (GA-BP) neural | | The system consists of a wind turbine, a PV | |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Guo et al., 2016) | Nil | Yes | Nil | Nil | Yes | Yes | the consumption of diesel fuel. | | Power balance constraints | high/low SOC constraints. Minimum start-up time of the desalination system. | network-based wind speed forecasting method | MATLAB/ RSCAD | panel, and a battery bank. | Yes |
| (Kamjoo et al., 2016) | Yes | Yes | Nil | Nil | Yes | Nil | Nil | Minimise the system total cost and maximum reliability index | Power balance constraints | Min/Max State-Of charge (SOC) constraints. Min/Max output power of wind and Solar PV. | Non-dominated Sorting Genetic Algorithm (NSGA-II) | Not restricted | Case studies | No |
| (Ogunjuyigbe et al.,2016) | Yes | Yes | yes | Nil | Yes | Yes | Nil | Minimise the system total cost and Emission | Power balance constraints | Min/Max State-Of charge (SOC) constraints. Min/Max output power of wind and Solar PV. | GA | Not specified | PV system, wind turbine system, battery storage, and 3-split small-rated Diesel generators | No |

## 2.4 Multi-area wind-thermal economic dispatch

The multi-area/interconnected power system focuses on moving the electricity from one region to another because the purchase and sale of electricity is a market-oriented agreement between two parties to reach an economically viable agreement (Zhu, 2015). The tie-line connects two or more separate areas of the power system, as shown in Figure 2.14.

The results of the literature review are compiled in Table 2.7 for the multi-area wind-thermal economic dispatch. Figure 2.15 represents Table 2.7 visually. This figure represents the number of papers per year.

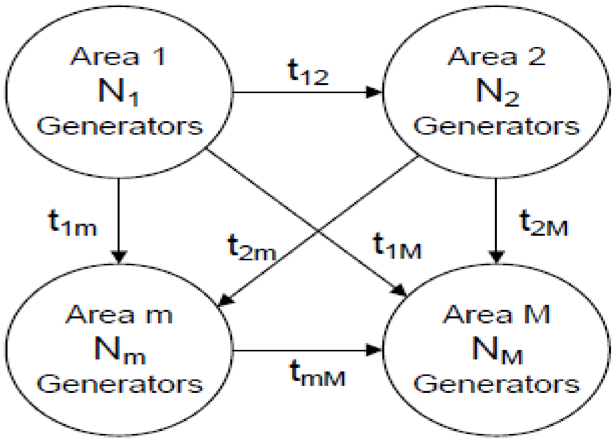



**Figure 2.14:** Interconnected power system(Krishnamurthy & Tzoneva, 2016)

**Table 2.7:** Quantity of papers for the multi-area dispatch problem on a yearly basis

| Number of publications year-wise | | |
|---|---|---|
| **Reference** | **Year of publication** | **Number of publications** |
| (Wang & Shahidehpour.,1992) | 1992 | 1 |
| (Streiffert, 1995) | 1995 | 1 |
| (Jayabarathi, 2005) | 2005 | 1 |
| Jeyakumar et al., 2006), | 2006 | 1 |
| ( Wang & Singh., 2007). | 2007 | 1 |
| (Manoharan et al., 2009), (Manoharan et al., 200ba), (Wang, & Singh., 2009), (Fesanghary & Ardehali.,2009) | 2009 | 4 |
| (Chen & Wang., 2010). | 2010 | 1 |
| (Sharma et al., 2011), (Somasundaram & Swaroopan., 2011) | 2011 | 2 |
| (Basu, 2013), (Soroudi, & Rabiee., 2013), and (Krishnamurthy., 2013) | 2013 | 3 |
| (basu, 2014) | 2014 | 1 |
| (Jadoun et al., 2015), (Secui et al., 2015),and (Pandit et al., 2015) | 2015 | 3 |
| (Ghasemi et al., 2016) | 2016 | 1 |

| | | |
|---|---|---|
| (Basu., 2019) | 2019 | 2 |
| (Chaudhary et al.,2020) | 2020 | 1 |
| (Sakthivel et al., 2021) | 2021 | 1 |
| (Ahmed et al.,2022) | 2022 | 1 |
| (Lotfi.,2023) | 2023 | 1 |
| Wang et al.,(2025), | 2025 | 1 |

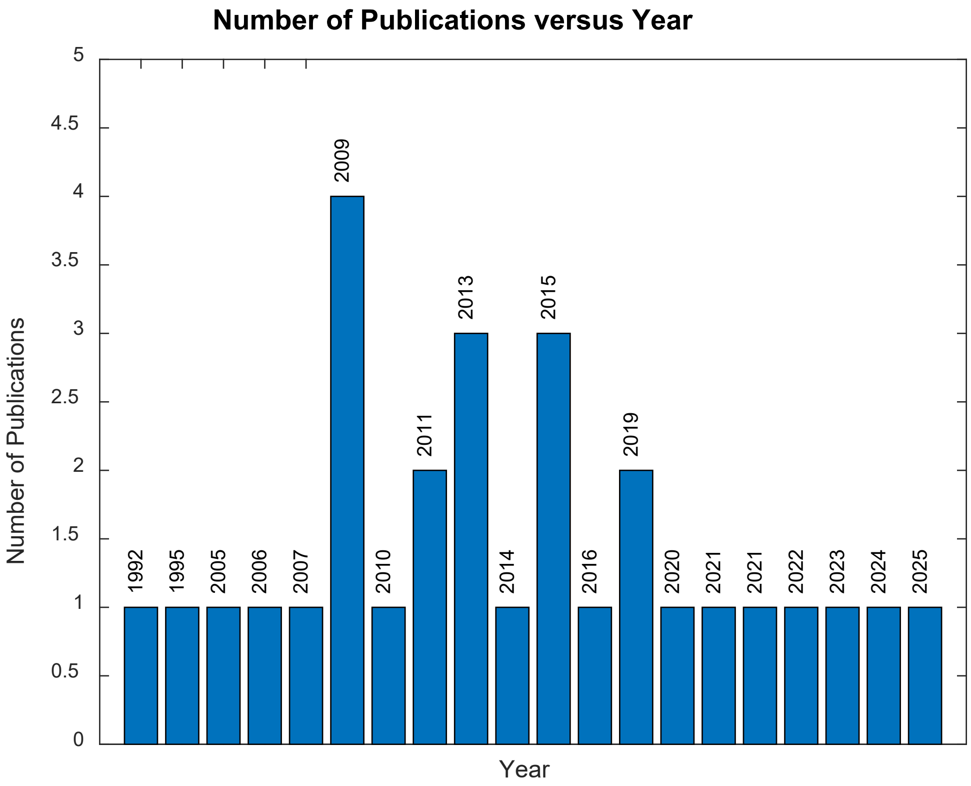



**Figure 2.15:** **Quantity of papers for the multi-area wind-thermal dispatch problem on a yearly basis.**

**Table 2.8:** **Techniques and algorithms applied to solve the multi-area wind-thermal economic dispatch problem**

| **Most used methods and algorithms** | | **Reference paper** | **No. of publications** |
|---|---|---|---|
| **Algorithm** | **Abbreviation** | | |
| Artificial bee Colony | ABC | (Basu, 2013) | 1 |
| Chaotic Global Best Artificial Bee Colony algorithm | CGBABC | (Secui et al., 2015), | 1 |
| Covariance Matrix Adapted Evolution Strategy | CMAES | (Manoharan et al., 2009) | 1 |
| Differential Evolution | DE | Manoharan et al., 2009), (Sharma et al., 2011), | 2 |

| | | | |
|---|---|---|---|
| Dynamically Controlled Particle Swarm Optimisation | DCPSO | (Jadoun et al., 2015). | 1 |
| Evolutionary Programming | EP | (Jayabarathi., 2005) | 1 |
| Evolutionary Programming with Levenberg-Marquardt Optimization | EP-LMO | (Manoharan et al., 2009b) | 1 |
| Expert systems | ES | (Wang & Shahidehpour., 1992) | 1 |
| Fuzzified Particle Swarm Optimisation | FPSO | (Wang & Singh., 2007) (Somasundaram & Swaroopan., 2011) | 2 |
| Harnomy Search | HS | (Fesanghary & Ardehali.,2009) | 1 |
| Hybrid Differential Evolution Particle Swarm Optimisation | DEPSO | (Ghasemi et al., 2016) | 1 |
| Multi-Objective Particle Swarm Optimisation | MPSO | (Wang & Singh, 2009). | 1 |
| Network Flow Programming | NFP | (Streiffert, 1995) | 1 |
| Optimality Condition Decomposition | OCD | (Soroudi, & Rabiee., 2013), and (Krishnamurthy, 2013) | 2 |
| Particle swarm optimisation | PSO | (Jeyakumar et al., 2006), (Chen & Wang, 2010). (Sharma et al., 2011), Manoharan et al., 2009) | 4 |
| Real-coded Genetic Algorithm, | RGA | (Manoharan et al., 2009) | 1 |
| Teaching-Learning-Based Optimisation | TLBO | (Basu, 2013) | 1 |
| Nondominated Sorting Genetic Algorithm-II (NSGA-II) | NSGA II | (Basu., 2019) | 1 |
| Chaotic Fast Convergence Evolutionary Programming (CFCEP) | CFCEP | (Basu., 2019) | 1 |
| Salp Swarm Algorithm (SSA) | SSA | (Chaudhary et al.,2020) | 1 |
| Multi-Objective Squirrel Search Algorithm (MOSSA) | (MOSSA) | (Sakthivel et al., 2021) | 1 |
| Crow Search Optimization algorithm (CSOA) | CSOA | (Ahmed et al.,2022) | 1 |
| Modified Grasshopper Optimization (MGO) | (MGO) | (Lotfi,2023) | 1 |
| Semi-Definite Programming(SDP) | SDP | (Alli,.2024) | 1 |
| Fractional order refined comprehensive learning (FORCL) LSHADE | (FORCL) LSHADE | (Wang et al.,2025), | 1 |

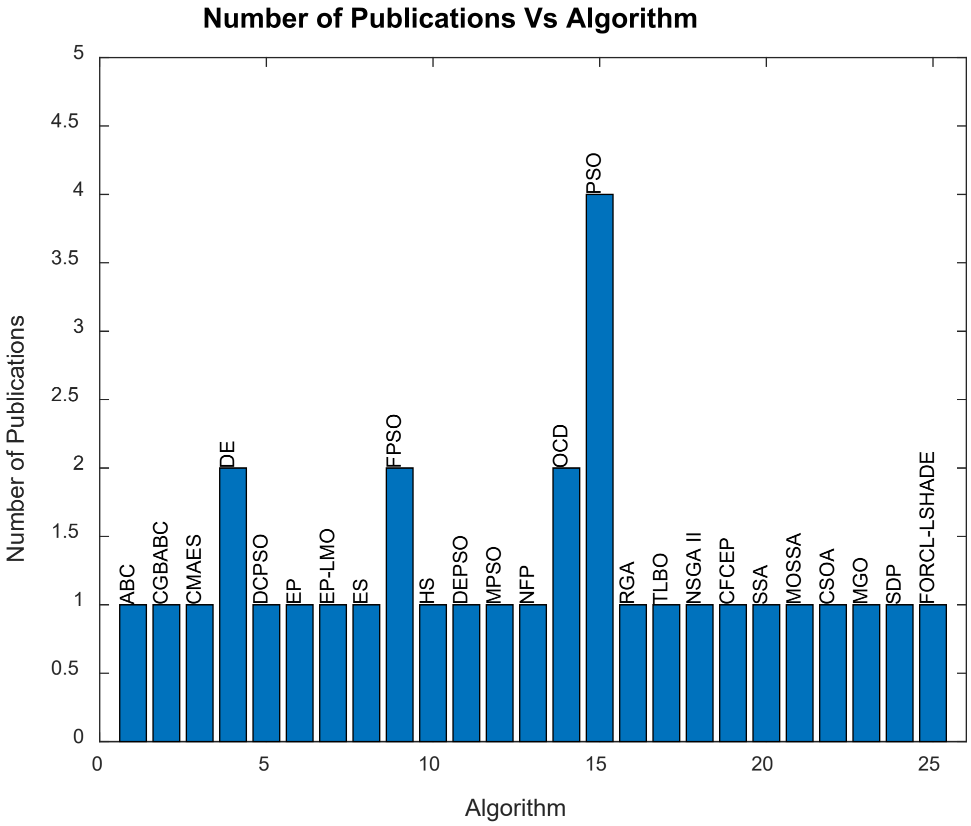



**Figure 2.16**: **Quantity of papers for the multi-area wind-thermal dispatch problem versus the algorithm**

### 2.4.1 Study on mathematical models and techniques applied to resolve the multi-area wind-thermal economic dispatch

#### 2.4.1.1 Single-criteria

Many researchers have solved the multi-area economic dispatch problem using the incremental cost method, decomposition method, incremental network flow, etc. In a medium-sized power system, solution times may be fast and efficient. However, in a larger, interconnected power system, the solution space and computational burden may increase due to the higher dimensionality of the optimization problem. To reduce computational burden, new hybrid optimisation methods are used.

The multi-area dispatch problem is solved using incremental network flow programming in (Streiffert, 1995). The tie line between two areas in a multi-area power system, along with the power balance equality and inequality constraints of a conventional generator, is formulated. In the work of (Sharma et al., 2011), the dispatch problem is formulated as a multi-area reserve-constrained problem. The objective function included both generator and transmission line power flows, subject to the linear area power balance excluding losses, and nonlinear constraints such as generating limits, ramp-rate limits, prohibited zones, tie-line limits, and area-wise spinning reserves. PSO and DE optimisation methods are used to solve the MAED, and the quality of the solutions is compared with that reported in the existing literature. The authors (Basu, 2013, 2014; Fesanghary & Ardehali, 2009) solved the MAED with the transmission losses included in the power balance constraints. The Artificial Bee Colony (ABC), Teaching-Learning-Based Optimization (TLBO), and Harmony Search (HS) algorithms were used to solve an optimization problem involving multiple fuels. A Dynamically Controlled Particle Swarm Optimization(DCPSO) is used to solve the multi-area dispatch problem, and is formulated with the quadratic fuel cost with valve point loading effect, area power balance is considered without transmission losses, generator limits, and nonlinear area spinning reserves constraints, and multi-area tie-line security limits are considered in (Jadoun et al., 2015).

In the research work of the authors (Manoharan, Kannan, Baskar, et al., 2009; Manoharan, Kannan, & Ramanathan, 2009) The multi-area optimisation problem is solved using Real-Coded Genetic Algorithm (RGA), Particle Swarm Optimization (PSO), Differential Evolution (DE), and Covariance Matrix Adapted Evolution Strategy (CMAES). Their solutions are compared with the Direct Search Method (DSM) and the Hopfield Neural Network (HNN), Evolutionary Programming (EP), and the Nelder–Mead simplex (NMS). The optimum values of the PSO method are validated using the Karush–Kuhn–Tucker (KKT) condition and Evolutionary Programming with Levenberg-Marquardt Optimization (EP-LMO). The improved Hopfield neural network approach is adopted to solve the economic dispatch problem of a multi-area system, considering power transfer security constraints, transmission losses, and power balance constraints (Yalcinoz & Short, 1998). In the research work of the authors (Somasundaram & Swaroopan, 2011)PSO is amended with fuzzy logic to form a fuzzified particle swarm optimisation method to solve the security-constrained Multi-Area Economic Dispatch (MAED) in order to obtain an optimal pareto solution. The security constraints for tie-line, generator limits, and power balance constraints with transmission losses are considered in their research work.

#### 2.4.1.1 Multi-criteria

Recently, hybrid optimisation techniques have gained popularity. In (M. Ghasemi et al., 2016), the authors integrated Differential Evolution (DE) and Particle Swarm Optimisation (PSO) to develop a hybrid Differential Evolution Particle Swarm Optimisation (DEPSO) to address the Multi-Area Economic Load Dispatch (MAED).In (Secui, 2015), the Multi-Area Economic Load Dispatch (MAEED) problem is solved utilising the Chaotic Global Best Artificial Bee Colony Algorithm (CGBABC). The Multi-Objective Particle Swarm Optimisation (MPSO) technique is employed to determine the Pareto optimal solutions for the MAEED problem, which is intended to reduce both fuel cost and emissions(Wang & Singh, 2009). In the research work of authors (Wang

& Shahidehpour, 1992), and (Krishnamurthy & Tzoneva, 2016), the multi-area dispatch problem is explained using a decomposition technique. The multi-area dispatch problem is defined using a decomposition strategy. This includes a top (master problem) and bottom (sub-problems) formation of a two-level decomposed hierarchy as seen in Figure 2.17. Thus, the master problem is broken down into sub-problems, in which the master distributes load demands to the sub-problems, and each area returns its solution as a cost to the system. Thus, the system's minimum operating cost acts as the aggregator at this level. The sub-problems are the sub-areas of the power system network. The subproblem for every network incorporates system λ and provides the network λ, where λ is the tie-line coordinator's variable vector.

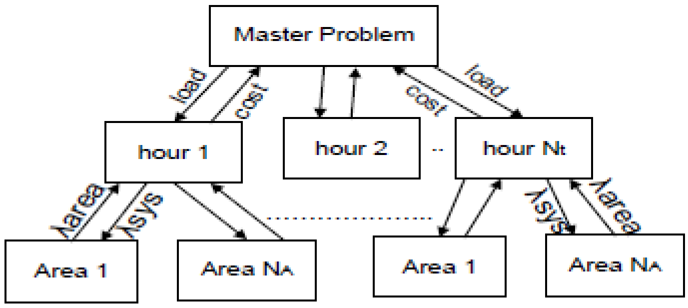



**Figure 2.17: Decomposition structure with two levels**
(Krishnamurthy & Tzoneva, 2016)

The multi-area economic dispatch problem is extended as a Piecewise Quadratic cost Function with multiple fuels (PQCF) in (Jayabarathi et al., 2005; Jeyakumar et al., 2006) Operating zones (POZ) are identified, and the optimal solution is obtained using Particle Swarm Optimisation (PSO), which is also compared with results from the classical Evolutionary Programming (EP). Furthermore, PSO is used for the MAEED dispatch problem in (Ming Chen & Wang, 2010) where the tie-line and tie-loss constraints between the areas represent additional constraints on the problem.

In addition, emissions are included in the cost function to establish the Multi-Area Nonconvex Combined Economic Emission (MANCEED) dispatch problem (Pandit, Srivastava, et al., 2015; Sharma et al., 2011). Then, the Differential Evolution with Fuzzy Selection (DEFS) is implemented to address the MANCEED problem, using a fuzzy framework. The mutation and crossover processes are optimised to reduce the complexity of multi-attribute decision-making, and a penalty factor technique with a limitation incorporated into the objective function. In the work of Wang and Singh, the MANCEED is solved using the Fuzzified Multi-Objective Particle Swarm Optimisation (FMPSO) technique, which successfully discovered a collection of non-dominated solutions in a single run. Compared with other techniques such as weighted aggregation and Multi-Objective Evolutionary Algorithms (MOEAs), it is far more effective at obtaining a global optimal solution.

The authors (Soroudi & Rabiee, 2013) published a paper on the multi-area dispatch problem, in which the objective function includes thermal, hydrothermal, and wind units, as well as a power-pool market for each area. The Optimality Condition Decomposition (OCD) method is used to solve this power system. The MAED problem is first decomposed into simple sub-problems, each representing a single area. This results in less computation time. Parallel computing is used to improve the computational time required to solve the MAED dispatch optimisation problem.

In (Basu, 2019a)The optimisation problem for MAWTEED is formulated using the power balance and inequality constraints, including power losses, tie-line power limits, and minimum and maximum power limits for hydro, thermal, and wind power plants. The problem is presented as a nonlinear optimisation with a non-smooth quadratic cost function and ramp-rate constraints. A Nondominated Sorting Genetic Algorithm-II (NSGA-II) is used to solve the problem. The method's performance is tested across a network of four Reservoir Hydro Plants in a multi-chain cascade, along with ten conventional power plants and two wind power plants in each region. Each region has its own load demand, and regions are interconnected through tie lines with other regions.

The same author (Basu, 2019b), extended the multi-area problem now including solar in the cost function and tested the effectiveness of chaotic fast convergence evolutionary programming (CFCEP) first using a three-area test system, that consists of a multi-chain cascade of four reservoir hydro plants, one wind power, solar PV, ten thermal power plants with Valve Point Loading Effect(VPLE) and one pumped hydro energy storage power plants. Secondly, using forty generating units and dividing the power system into four areas with different power demands of 15%, 40%, 40%, and 15% of the maximum demand of 10500MW. Lastly, using a 140-unit test system with a load demand of 49 342MW.

A Salp Swarm algorithm(SSA) is used to solve the MAWTEED problem where the effectiveness of the optimisation method is verified with different case studies, first case consists of 4 areas with four generators each, second has forty units divided into four areas and third has 2 areas with 20 generators each, forth has two area network that consists of forty units with 20 units in each area, and three wind thermal having 110MW. In (Ahmed et al., 2022), the multi-area power system is formulated as a power balance problem with inequality constraints and tie-line limits. The problem includes thermal, wind, and solar generating units. The problem uses the Crow Search Optimization

(CSOA) algorithm. This optimization algorithm is validated using different test systems, as illustrated in Table 2.9 below.

(Wang et al., 2025) formulated an MAWTEED problem with thermal, wind, and solar power systems. The power balance and inequality constraints are included in the constraints. A WDF and a lognormal distribution are used to model uncertainty in wind power and solar power, respectively. A Fractional order refined comprehensive learning (FORCL) LSHADE is used to determine the optimal values of fuel and emissions during the optimization process. The algorithm is validated with a test system having 3 areas with 10 units, including VPE, MFO, and the following cases are examined: case(i) No solar and wind power, only thermal generation, case (ii) Both solar and wind power are considered. Test system II has four areas and 40 generating units, 10 units in each area, and the following cases are considered: case (i) No Solar and Wind Power units. Case(ii) Both Wind and Solar are included. Test System III includes a practical provincial power system in China with 46 units. The case includes one without wind or solar power, and the latter includes both.

(Sakthivel et al., 2024) developed a Multi-Objective Squirrel Search Algorithm (MOSSA) to solve a multi-objective optimisation problem with multiple fuels and valve-point effects. The problem is validated through different case studies: one with 10 units and three areas, another with 40 units and four area systems, and, lastly, a complex 140-unit Korean power system that considers valve-point effects and multiple fuels.

(Alli, 2024) used the Semi-Definite Programming (SDP) method to solve a multi-area wind-thermal economic dispatch problem. Both equality and inequality constraints are included in the nonconvex optimization problem. The algorithm's effectiveness is first tested on a two-area, six-unit system with three generators in each area, without wind power. Secondly, in a two-area system with 40 generating units (20 in each area), without wind power. Lastly, tested on a two-area system with 40 generating units in each area, three conventional units are replaced by identical wind power units of 110MW each.

In (Lotfi, 2022), a multi-objective optimization problem is solved using the Modified Grasshopper Optimization (MGO) algorithm. The power balance and inequality constraints are considered as presented in Table 2.9. A 10- and 40-unit system with 3 zones is used to validate the algorithm. The algorithm is then contrasted with other heuristic algorithms reported in the survey.

### 2.4.2 Multi-area economic emission dispatch problem formulation

The goal of Multi-Area Wind-Thermal Dynamic Economic Dispatch (MAWTDEED) is to reduce total emissions from thermal units; limit the power outputs of thermal units and wind farms in each region; and allow power transfer among power generation regions. This is mathematically formulated (Ghasemi et al., 2016; Hetzer et al., 2008; Ming Chen & Wang, 2010; Secui, 2015).

$$F_{mai}(P_{gi}, P_w, E_i) = \sum_{ma=1}^{M}\sum_{i=1}^{n} F_{wmai} + F_{Tmai} \qquad [\$/h] \qquad (2.64)$$

Where is the number of online generators for the area $ma$ is represented by $n$ in a $M$ area generation unit. $F_{wmai}$ is the total amount of power that the wind generating units can produce. $F_{Tmai}$ is a combined economic emission dispatch objective function provided by equation (2.60). The following equality and inequality limitations bind the objective function. The overall power balance must equal the demand for power per area, including power flows to and from other areas. This is mathematically expressed as:

$$\sum_{i=1}^{n} P_{mai} - \sum_{\substack{m=1 \\ m\neq k}}^{M} \left[t_{mak} - (1-\rho_{kma}) \times t_{kma}\right] - P_{D,ma} - P_{L,ma} = 0 \qquad [\$/h] \qquad (2.65)$$

In this particular instance $P_{mai}$, represents the output power from combined thermal and wind generators, as shown by equation (2.64). The following are the tie-line limits:

$$t_{kma}^{\min} \le t_{kma} \le t_{kma}^{\max} \qquad (2.66)$$

Where

$t_{kma}$ Power transfer through a tie-line

$\rho_{kma}$ Transfer loss ratio through the tie-line

$t_{kma}^{\max}$ and $t_{kma}^{\min}$ Maximum and Minimum power transfer capacity limits through the tie-line

Other limitations are similar to those mentioned in the previous formulation of the single-area dispatch problem. Table 2.9 summarises the review for single, multi-objective, and multi-area economic dispatch.

**Table 2.9:** **Multi-area review papers for single and multi-criteria dispatch problem**

| Reference | System Component | | | | Objective function | | Constraints | | Multi-area constraints | Algorithm | Software | Power system considered | Real-time implementation |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Wind | PV | Hydro | Diesel | Single | Multi | Equality | Inequality | | | | | |
| (Streiffert., 1995) | Nil | Nil | Nil | Yes | Fuel cost minimisation | Nil | Real power balance with transmission line constraints. | Minimum and maximum power limits for a conventional generator. Minimum and maximum tie line limits | Min/Max tie line flow limit | Network flow programming | Not mentioned | T<br>Three case studies were done on a four-area network system with four generators. | **No** |
| (Sharma et al., 2011) | Nil | Nil | Ni | Yes | Fuel cost Minimisation with valve point loading effect | Nil | Real power balance | Generating limits. Ramp rate limit. Prohibiting zones. Area-wise spinning reserves | Min/Max tie line flow limits | Particle swarm optimisation(PSO) and Differential evolution(DE) | MATLAB | Two area networks, each with two generators.<br>Four area power system with four generators in each area.<br>Two-area, 40-unit system. | No |
| (Basu, 2013) | Nil | Nil | Nil | Yes | Fuel cost Minimisation with valve point loading effect with multiple fuels | Nil | Real power balance with transmission line constraints. | Generating limits. Ramp rate limit. Prohibiting zones. Area-wise spinning reserves | Min/Max tie line flow limits | Artificial Bee Colony(ABC) | MATLAB | Three different test systems were used.<br>(i) The first system consists of two areas with three generators in each area, with prohibiting zones. Transmission line considered<br>(i) The second system consists of three areas, with area 1 having 4 units, area 2 having 3, and area 3 having the last 3. | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | (iii) The third system consists of four areas; each area consists of 10 units to make a total of 40 generating units. | |
| (Basu, 2014) | Nil | Nil | Nil | Yes | Fuel cost Minimisation with valve point loading effect with multiple fuels | Nil | Nil | Generating limits. Ramp rate limit. Prohibiting zones. Area-wise spinning reserves | Min/Max tie line flow limits | Teaching-Learning-Based Optimisation( TLBO) | MATLAB | Three different test systems were used. The first system comprises two areas, each with three generators, and includes prohibiting zones. The second system has three areas: area 1 has 4 units, area 2 has 3, and area 3 has the last 3. (iii) The third system consists of four areas; each area consists of 10 units to make a total of 40 generating units. | No |
| (Jadoun et al., 2015). | Nil | Nil | Nil | Yes | Fuel cost Minimisation with valve point loading effect. | Nil | Nil | Generating limits. Ramp rate limits. Area-wise spinning reserves | Min/Max tie line security limits | Dynamically Controlled Particle Swarm Optimisation( DCPSO) | MATLAB | Three case studies are considered. Case study 1 has two areas with two generators each. Case study 2 has 4 areas with 10 generators each. Case study 3, with, consists of 5 areas with 28 generators in each area. | No |
| (Manoharan et al., 2009) | Nil | Nil | Nil | Yes | Fuel cost Minimisation with multiple fuels | Nil | Real power balance with the | Generating limits. Ramp rate limit. | Min/Max tie line power flow limits | Real-coded Genetic Algorithm (RGA), Particle Swarm | MATLAB | Case 1: 4-unit system with two areas. Case 2: 10-unit system including multi-fuel options with three areas. | No |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | transmission line | Area-wise spinning reserves | | Optimization (PSO), Differential Evolution (DE), and Covariance Matrix Adapted Evolution Strategy (CMAES) | | Case 3: 120-unit large-scale system with two areas. | |
| (Ghasemi et al., 2016) | Nil | Nil | Nil | Yes | Nil | Fuel cost Minimisation with valve point loading effect with multiple fuels and emissions | Real power balance with transmission line constraints | Generating limits. Ramp rate limit. Area-wise spinning reserves. Prohibiting zones | Min/Max tie line power flow limits. | Differential Evolution Particle Swarm Optimisation( DEPSO) | MATLAB | 4 area systems with 16 generators and two area networks with 40 units. | No |
| (Secui.,, 2015) | Nil | Nil | Nil | Yes | Nil | Fuel cost Minimisation with valve point loading effect with multiple fuels and emissions | Real power balance with transmission line constraints | Generating limits. Ramp rate limit. Area-wise spinning reserves. Prohibiting zones | Min/Max tie line power flow limits. | Chaotic Global Best Artificial Bee Colony algorithm(CGBABC) | Mathcad | (i) 3 area systems with 10 generators.<br>(ii) 4 areas network with 40 units. | No |
| (Wang, & Singh., 2009) | Nil | Nil | Nil | Yes | Nil | Fuel cost Minimisation and emission. | Real power balance constraints | Generating limits. Ramp rate limit. Area-wise spinning reserves. | Min/Max tie line power flow limits. | Mult-Objective Particle Swarm Optimisation( MOPSO) | MATLAB | Four areas with 4 generators in each area | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Prohibiting zones | | | | | |
| (Wang & Shahidehpour.,1992) | Nil | Nil | Nil | Yes | Fuel cost function minimisation | Nil | Real power balance constraints. | Generating limits. Ramp rate limit. Area-wise spinning reserves. Prohibiting zones | Tie–line active power flow limits | Expert systems | Not Mentioned | 4 area system with 26 units in each area | No |
| (Manoharan et al., 2009) | Nil | Nil | Nil | Yes | Fuel cost Minimisation with multiple fuels | Nil | Real power balance with transmission line constraints | Generating limits. Ramp rate limit. Area-wise spinning reserves | Min/Max tie line power flow limits | The Evolutionary Programming with Levenberg-Marquardt Optimization (EP-LMO) | MATLAB | Case 1: 4-unit system with two areas. Case 2: 10-unit system including multi-fuel options with two areas. | No |
| (Jeyakumar et al., 2006), | Nil | Nil | Nil | Yes | Fuel cost Minimisation with multiple fuels. Four different examples to illustrate the effectiveness of PSO, with MAED, MAPQCF, MACEED, and MAED with prohibiting zones | Fuel cost Minimisation with multiple fuels and emissions | Real power balance with the transmission line | Generating limits. Prohibiting zones | Min/Max tie line power flow limits | Particle Swarm Optimisation( PSO) | MATLAB | (i) Four-area system interconnected by six tie lines. (ii) Ten generating units, each with an area of four areas, with three types of fuel. (iii) Six generating units, considering emission and transmission (iv) a fifteen-unit power system with four of the units having up to three prohibited operating zones. | No |
| | | | | | Fuel cost reduction with multiple fuels. Four different examples to | Fuel cost | Real power | | | | MATLAB | (i) Four-area system interconnected by six tie lines. (ii) Ten generating units, each with an | |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Jayabarathi, 2005) | Nil | Nil | Nil | Yes | illustrate the effectiveness of PSO, with MAED, MAPQCF, MACEED, and MAED with prohibiting zones | reduction with multiple fuels and emissions | balance with transmission line constraints | Min/Max generator limits. Prohibiting zones | Power flows from one area to other minimum and maximum values | Evolutionary Programming (EP) | | area of four areas, with three types of fuel. (iii) Six generating units, considering emission and transmission (iv) a fifteen-unit power system with four of the units having up to three prohibited operating zones | No |
| (Somasundaram & Swaroopan., 2011) | Nil | Nil | Nil | Yes | Minimise fuel cost | Nil | Power balance and Transmission losses constraints | Min/Max generator limits. | Tie-line power flow limits | Fuzzified Particle Swarm optimisation | Not Mentioned | Three-area 32-bus system. | No |
| (Fesanghary & Ardehali.,2009) | Nil | Nil | Nil | Yes | Fuel cost minimisation | Nil | Power balance and Transmission losses constraints | Min/Max generator limits. Prohibited zones. Ramp rate limit | Tie-line power flow limits | Harmony Search (HS) algorithm | Visual C++ | Six different case studies are studied. (i)Standard MAED with 38 units. (ii)ED problem with valve-point loading effect with 13 generating units. (iii)ED problem with prohibited operating zones with 15 units MAED with tie line flow limits in four area network. (iv)ED problem with ramp-rate limits, with 6 units having ramp rate and prohibited zone limits. | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Chen & Wang, 2010). | Nil | Nil | Nil | Yes | Nil | Fuel cost reduction and emission | Power balance Transmission losses | Min/Max generator | Tie-line power flow limits. | PSO | MATLAB | EEE 30-bus system with 6 generators | No |
| (Pandit et al., 2015), | Nil | Nil | Nil | Yes | Nil | Minimise fuel and emission | Power balance Transmission losses | Prohibiting zones. Area-wise spinning reserves. Generator capacity constraints | Tie-line power flow limits. | Differential Evolution with Fuzzy Selection (DEFS) | MATLAB | Four area network system with 16 generators. | No |
| (Wang & Singh., 2007) | Nil | Nil | Nil | yes | Nil | Fuel cost reduction and emission | Power balance. Transmission losses | Generator capacity constraints | Tie-line power flow limits. | Fuzzified Multi-Objective Particle Swarm Optimisation( FMOPSO) | C++ | IEEE 118-bus test system with 14 generators | No |
| (Soroudi, & Rabiee., 2013) | Yes | Nil | Yes | Yes | Fuel cost minimisation | Nil | Power balance. The purchased power from the pool market | Wind and thermal generator limits. reservoir volume min/max constraints | Tie-line power flow limits. | Optimality Condition Decomposition(ODC) | GAMS environment solved by the CONOPT solver | Case I: three interconnected systems with assumed power generated by wind, hydro, and the power pool. Case II used five interconnected areas, with given power generation from wind, hydro, and pool market | Yes |
| (Basu.,2019) | Yes | Nil | Yes | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind and thermal generator limits. reservoir volume min/max constraints | Tie-line power flow limits. | Nondominated Sorting Genetic Algorithm-II (NSGAII) | MATLAB | Comprises a multi-chain cascade of four reservoir hydro plants, ten conventional, and two wind power energy systems in each area. | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Basu.,2019) | Yes | Nil | Yes | Yes Solar and thermal | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind and thermal generator limits. reservoir volume min/max constraints, ramp rate constraints | Tie-line power flow limits. | Chaotic Fast Convergence Evolutionary Programming (CFCEP) | MATLAB | Comprises a multi-chain cascade of four reservoir hydro plants, ten conventional, one wind, and solar power energy systems in each area. A four-area, 40 units with ten generators in each area. 140 unit 12 areas and 12 generators in each area | |
| (Chaudhary et al.,2020) | Yes | Nil | Nil | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind and thermal generator limits. | Tie-line power flow limits. | Salp Swarm algorithm(SSA) | MATLAB | 4 areas with 4 generators each, the second has forty units divided into 4 areas and third has 2 areas with 20 generators each, the fourth has two area networks that consist of forty units with 20 units in each area, and three wind thermal have 110MW each | No |
| (Sakthivel et al., 2021) | Nil | Nil | Nil | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | thermal generator limits. | Tie-line power flow limits | Multi-Objective Squirrel Search Algorithm (MOSSA). | MATLAB | 10 units and three areas, another with 40 units and four areas, and a 140-unit Korean power system. | |
| Ahmed et al.,2022 | Yes | Yes | Nil | Yes | Nil | Fuel cost **reduction and** | Power balance. | Wind, thermal, and solar | Tie-line power flow limits. | Crow Search | MATLAB | 10 units allocated on three areas, Area1 has 4 committed units, | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | **emission** | Transmission losses | generator. min/max constraints, ramp rate constraints, Prohibited Operating Zones (POZs),tie-line constraints | | Optimization algorithm (CSOA) | | Area 2 has 3 conventional units, and Area 3 has 3 as well. This test system has 3 different case studies. Test system II has 40 generating units, and 10 generators in each area. Power demand differs in each area. Test system III, consists of 140 units across the country, where 1–40 units are thermal (mostly coal powered), 41–91 are gas powered plants, 92–111 are nuclear powered plants, and 112–140 They are oil-powered plants. Test system IV has 5 thermal units integrated with two wind turbines and two solar PV systems. | |
| (Lotfi.,2023) | Yes | Nil | Nil | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind and thermal generator. min/max constraints. Rampe rate,multi-Fuel option, and Prohibited zones for thermal units. | Tie-line power flow limits. | Modified Grasshopper Optimization (MGO) algorithm | MATLAB | two systems with a 10-unit, 3-zone test system and a 40-unit 3-zone test system, | No |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| (Alli.,2024) | Yes | Nil | Nil | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind, and thermal generator. min/max constraints. Rampe rate and prohibited zones of thermal generators | Tie-line power flow limits. | Semi-Definite Programming (SDP) | | Two area, six-unit system with three generators in each. Two-area system with 40 generating units, 20 generators in each area. Two-area system with 40 generating units in each area, three of the conventional units are replaced by identical wind power units of 110MW each. | No |
| Wang et al.,(2025) | Yes | Yes | Nil | Yes | Nil | Fuel cost **reduction and emission** | Power balance. Transmission losses | Wind, thermal and solar generator. min/max constraints, | Tie-line power flow limits. | Fractional order refined comprehensive learning (FORCL) LSHADE | MATLAB | Test system having 3-areas with 10-units.Test system II, has 4 areas and 40 generating units, 10 units in each area, Test System III, includes a practical provincial power system of China having 46-units. | No |

## 2.5 Outcomes from analysis of the literature

Numerous articles have been reviewed from 1945 to 2026 for single-area wind-thermal, from 2005 to 2025 for single-area hybrid systems, and from 1992 to 2025 for the multi-area wind-thermal economic dispatch problem. Tables 2.3, 2.6, and 2.9 provide the review summaries for these articles. The review emphasises two crucial topics: the algorithm used for addressing a particular type of dispatch problem and the formulation of wind-thermal problems under system inequality and equality constraints.

### 2.5.1 Examination of the problem formulation applied for solving both single-area and multi-area economic dispatch problems with single and multi-criteria.

The objective function defines the single- and multi-area dispatch problem. For thermal generators, a quadratic function with valve-point loading has been adopted from many recent works, as it's a more realistic approach than a linear-quadratic function. For wind turbine generators, direct, overestimation, and underestimation cost functions have been utilized. The direct one applies to the wind farm, as per ownership. For assessing the emission cost function, a second-order quadratic function, an exponential function, and a linear-quadratic equation can be used. The total area power demand is obtained based on load, transmission losses (tie-line capacity restriction), nonlinear restrictions (prohibited zones), and the max/min output power of thermal and wind generators. The objective function of the multi-area wind-thermal economic dispatch problem is the summation of fuel, wind, and environmental costs, subject to constraints.

### 2.5.2 Examination of techniques and algorithms applied in solving single-area and multi-area economic dispatch problems with single-and multi-criteria.

The economic dispatch research is evolving towards the implementation of heuristic optimisation methods such as Evolutionary Iteration Particle Swarm optimisation(EIPSO) (Lee, 2007), Gravitational Acceleration Enhanced Particle Swarm Optimisation(GAEPSO)(Jiang et al., 2015), Hybrid Particle Swarm Optimisation (HPSO) (Niu&Wei,.2013), Series Particle Swarm(SPSO-DE) Differential Evolution Optimisation(Pandit, Chaudhary, et al., 2015b), Dynamic controlled Particle Swarm Optimisation(DCPSO)(Jadoun et al., 2015), Fuzzified Particle Swarm Optimisation(FPSO) (Wang & Singh, 2007), (Somasundaram & Swaroopan, 2011), and Differential Evolution Particle Swarm Optimisation(DEPSO)(Ghasemi et al., 2016) used as a tool to solve the power system optimisation problems. These optimisation methods are indicated in Figures 2.9, 2.12, and 2.15. Observing from these Figures that the grouping of several optimization techniques is commonly employed to find global optima and reduce the computational time required to solve the entire dispatch problems. A more frequently employed optimisation technique is particle swarm optimisation. This approach is applied to a wind-thermal economic emission dispatch problem in a multi-area, multi-objective system, where wind is considered an alternative power generation resource, with the relative reduction coefficient of power generation cost and the relative minimization coefficient of emissions. The multi-objective problem is transformed into a single objective using a weighting factor via separation and mutation, a method validated by various researchers in the literature.

## 2.6 Conclusion

The formulations of the multi-area economic dispatch problem have been discussed. The multi-area economic dispatch problem is formulated with a tie-line real-power controller, and the goal function can be formulated as a single criterion or a multi-criterion, depending on the research purpose. In addition, the multi-area economic dispatch problem is solved by various approaches and algorithms. Heuristic optimization approaches are employed (DEPSO), Differential Evolution Particle Swarm Optimization (Ghasemi et al., 2016). This means that a review is held regarding computation times to determine optimal solutions for different types of economic dispatch problems. The review shows that the MAWTEED problem involves many variables, nonlinear equations, and high dimensionality, resulting in very complex calculations. (Soroudi & Rabiee, 2013). The Optimality Condition Decomposition (OCD) (Krishnamurthy & Tzoneva, 2016; Soroudi & Rabiee, 2013) would allow the problem to be decomposed into a two-level subproblem. Distributed and parallel computing can minimize computation time when algorithms are distributed and parallelized. MAWTEED can be solved using the Lagrange Decomposition Coordinating Method, which is one of the best means of parallelizing a solution, as it decomposes the comprehensive problem into sub-problems whose solutions add up to the comprehensive solution. Thus, optimality is reached.

# CHAPTER THREE

## WIND-THERMAL ECONOMIC EMISSION DISPATCH(WTEED) PROBLEM USING LAGRANGE'S ALGORITHM

## 3.1 Introduction

Generation of electricity from fossil fuels pollutes the environment by releasing pollutants such as sulfur dioxide (SO2), Nitrogen oxides (NOx), and carbon dioxide (CO2). The objective of the wind-thermal economic emission dispatch problem in electrical power system generation is to meet the power system equality and inequality constraints while dispatching electrical power to the committed conventional and wind generating units to meet load demand over a specific period. Wind power facilities are crucial to electricity production. These Distributed Energy Resources (DERs) offer an opportunity to substantially reduce production costs and atmospheric pollution.

In this chapter, fuel cost and pollutant reductions are covered, with transmission losses accounted for using the B matrix. The fuel cost and emission quadratic models are expressed without including valve-point loading effects. Both wind and conventional generating unit constraints are considered. The Lagrange's methods, algorithms, and software for the solution of Wind-Thermal Economic Combined Emission Dispatch (WTEED) problems are developed in this Chapter.

The Wind-Thermal Economic Emission Dispatch problem is first formulated in this chapter. The obtained models are then used to develop software to optimize this power dispatch problem. Three experimental IEEE 30-bus systems: 6-unit, 10-unit, and 40-unit are applied to test the developed algorithm.

Different numbers of available wind-generating units are utilised on a cluster of wind farms. Later, the results obtained with the proposed algorithm are compared with those of the existing heuristic algorithm.

## 3.2 WTEED problem formulation

### 3.2.1 Wind-thermal economic emission formulation for a single area dispatch problem

The problem formulation for the wind-thermal economic dispatch problem employing a single criterion function is presented. The goal of the Wind-Thermal Economic Emission Dispatch (WTEED) is to reduce the overall operating costs and emissions of the entire generation system, subject to specific operational constraints. The overall cost for conventional and wind generators can be expressed mathematically as( Ghasemi et al., 2016).

$$F(P_g, P_w) = \sum_{i=1}^{N_G} F_i(P_{gi}) + \sum_{j=1}^{N_F} \sum_{k=1}^{N_W} W_{j,k}(P_{w_{j,k}}) \tag{3.1}$$

where, $N_G$ is the quantity of electrical generators, $N_F$ is the number of wind power farms, $N_w$ is the quantity of wind generators, $P_{gi}$ is the active power for electrical generators, and $F(P_g, P_w)$ is the total expense of all generating units. The wind generating units' output power is denoted by the final term in equation (3.1), $W_{j,k}(P_{w_{j,k}})$.The cost function for conventional generating units is given by $F_i(P_{gi})$. It is expressed as in equation (2) and is developed using a quadratic function without a valve-point loading effect(Kothari & Dhillon, 2012). Figure 3.1 demonstrates how the thermal cost function can be represented with or without valve-point loading. The thermal generator loading is illustrated using arrows, as in Figure 3.1.

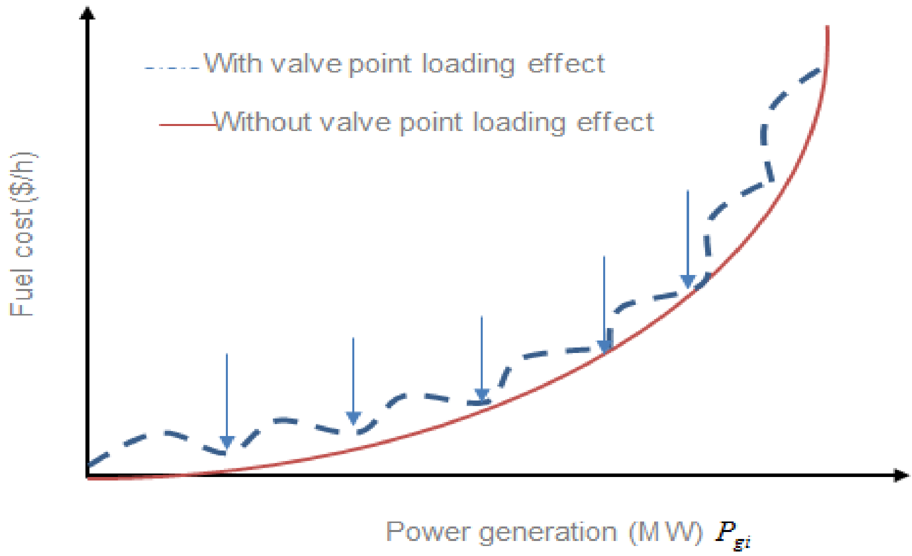



**Figure 3.1:** **Conventional cost curve with or without valve-point loading effect.**

$$F_i(P_{gi}) = a_i + b_i P_{gi} + c_i P_{gi}^2 \tag{3.2}$$

Where the fuel cost co-efficient of the $i^{th}$ generators are represented by $a_i, b_i$, and $c_i$ respectively. Equation (3.3) provides a further representation of the wind power production(A. Ghasemi et al., 2016).

$$W_{j,k}(P_{w_{j,k}}) = \phi_{D_{j,k}} P_{E_{j,k}} + \phi_{OE_{j,k}} \times (P_{A_{j,k}} - P_{E_{j,k}}) + \phi_{UE_{j,k}} \times (P_{E_{j,k}} - P_{A_{j,k}}) \tag{3.3}$$

where $P_{E_{j,k}}$, and $P_{A_{j,k}}$are the expected and actual/available wind power output of the turbine $j$ in a wind farm $k$ in [MW], $\phi_{D_{j,k}}, \phi_{OE_{j,k}}$, and $\phi_{UE_{j,k}}$ are the direct, overestimation, and underestimation cost coefficients of the turbine $j$ in a wind farm $k$ [$/MWh].

#### 3.2.1.1 The emission cost function formulation

Industrial advancement has caused a significant increase in pollutants to the atmosphere.

When coal is used in the thermal plant utilities, harmful gaseous pollutants such as carbon oxides($CO_2$), sulphur oxides($SO_2$), and nitrogen oxides($NO_x$) are discharged. The Green Power committee suggests using low-emission sources to address the economic dispatch problem. Therefore, there is a need for improved control techniques that can achieve lower pollution levels at a realistic fuel cost. Pollutants such as $CO_2$, $SO_2$ and $NO_x$The emissions by conventional generators should be incorporated in the optimisation objective function. Wind generators are not exposed to emissions, and the output power of the wind turbines depends on the wind velocity, so the emission cost for these generators is zero. The total emission of these impurities is the sum of the quadratic cost function and the emission values, multiplied by the exponential function of the valve-point loading effect. This can be expressed mathematically as (Abdelaziz et al., 2016)

$$E_T = \sum_{i=1}^{N_E} E_i(P_i) = \sum_{i=1}^{N_E} \quad (\alpha_i + \beta_i P_i + \gamma_i P_i^2) \tag{3.4}$$

where $E_T$ is the total emission in [kg/h], and the emission coefficients are $\alpha_i, \beta_i, \gamma_i$, for generating units in [kg/h].

**(a) Constraints for inequality and equality**

(b) Constraints for real power balance

The total of the power demand and transmission line losses equals the output power of the thermal and wind generation units. It can be indicated mathematically as shown in equation (3.5).

$$\sum_{i=1}^{N_G} P_{gi} + \sum_{k=1}^{N_W} P_{wh_{j,k}} = P_D + P_{Loss} \tag{3.5}$$

where, $P_D$ and $P_{Loss}$ are the power system overall load demand, and transmission line losses. Transmission losses in electric power systems are primarily caused by the distribution of electric power over lengthy distances or in vast areas with relatively low load density. Transmission losses must be incorporated into the formulation of the wind-thermal dynamic economic dispatch problem. In (Huang et al., 2018)The transmission loss formula, subsequently termed George's formula, is given by the first quadratic term in equation (3.6). The equation is also known as the loss formula or the B-coefficient technique. A linear term and a constant are added to the quadratic expression to get an improved transmission loss equation(Huang et al., 2018), and is termed Kron's formula and is given in equation (3.6)(Zhang et al., 2019). The B-coefficient technique allows coordination of power losses by arranging each plant's output to increase economy for a specified load.

$$P_L = \sum_{m=1}^{N_G+N_W} \quad \sum_{m'=1}^{N_G+N_W} P_m B_{mm'} P_{m'} + \sum_{m=1}^{Ng+Nw} B_{om} P_m + B_{00} \tag{3.6}$$

where the wind-thermal generator index is given by $m$ and $m'$ and power loss coefficients given by $B_{mm'}$,′ $B_{0m}$ and $B_{00}$respectively.

The following inequality constraints are applicable to the objective function.

**Inequality Constraints**

Generation capacitor for conventional and wind power constraints
Equations (3.7) and (3.8) provide the operational output power of each generating unit, which is constrained by its minimum and maximum operating capacities.

$$P_{gi}^{min} \leq P_{gi} \leq P_{gi}^{max}, i = 1,2, \dots N_g \tag{3.7}$$

$$0 \leq P_{j,h} \leq P_{R,j,h} \quad j = 1,2, \dots N_w \tag{3.8}$$

where the maximum and minimum output power of the $i^{th}$ thermal is represented by ${P_{gi}}^{max}$ and ${P_{gi}}^{min}$, and the rated power of $j^{th}$ wind output generator given by $P_{R,j,h}$.

#### 3.2.1.2 Probability distribution function for wind power

Representing short-term wind speeds is crucial for monitoring wind energy potential. PDFs are mainly used to describe wind speed interpretations. The appropriateness of numerous PDFs has been studied in various regions worldwide. The selection of the PDF is crucial in wind energy investigation because wind power is formulated as a clear function of wind speed distribution parameters.

A PDF that better fits the wind speed data reduces uncertainty in approximations of wind power output. The Weibull two-parameter PDF and the Rayleigh PDF are the most frequently used probabilities in wind speed data analysis. Many have modelled wind speed using Weibull distributions to formulate the wind-thermal economic dispatch problem. Weibull PDFs offer significant advantages over other probability distributions. Among its significant advantages, it requires only two parameters to estimate, and its probability density distribution is positively skewed, favouring moderate wind speeds.

The PDF can be represented using the Weibull Probability Density Function (WPDF)(Masters, 2004) and is given in Equation (3.9).

$$f(V) = \frac{k}{c}\left(\frac{V}{c}\right)^{k-1} e^{-(V/c)^k} \quad 0 \leq V < \infty \tag{3.9}$$

Where k is the shape parameter (dimensionless), and c is the scale parameter (in [m/s]) at a particular location. Equation (3.10) presents the Cumulative Distribution function (CDF) based on equation (3.9) above.

$$F(V) = P(v \leq V) = \int_{V=0}^{V=x} \left(\frac{k}{c}\left(\frac{V}{c}\right)^{k-1} e^{-(V/c)^k}\right) dv = 1 - exp\left(-\left(\frac{V}{c}\right)^k\right) \tag{3.10}$$

The $n^{th}$ instant for any probability density function $f(V)$ is described by Equation (3.11) according to the PDF(Masters, 2004).

$$\overline{V^n} = \int_{V_{min}}^{V^{max}} f(V)dV = C^n\Gamma\left(\frac{n}{k}+1\right), n = 1,2 \ldots \qquad (3.11)$$

where Equation (3.12) represents the Gamma function $\Gamma$ and is given as

$$\Gamma(\beta) = \int_0^{\infty} x^{\beta-1}\, exp(-x)dx \qquad (3.12)$$

By using equations (3.9)-(3.12), the probability distribution of wind power can be derived. The link between the wind generation's output power and wind speed is now expressed for each power unit by using a straightforward linear piecewise function, and is provided in equation (3.13)(Leon-Garcia, 2008).

$$\begin{cases} P_w = 0, & v < v_C \text{ or } v > v_F \\ P_w = P_R \frac{v-v_C}{v_r-v_C}, & v_C \le v \le v_R \\ P_w = P_R, & v_R \le v \le v_F \end{cases} \qquad (3.13)$$

where the power delivered and rated power by a wind turbine in [MW] are given by $P_w$ and $P_R$, and cut-in, rated, and cut-out/ furling wind speed in [m/s] are known by $v_C, v_R$ and $v_F$ respectively. A continuous or a discrete probability function can be utilized to represent the wind power curve(Qu et al., 2016; Xian Liu & Xu, 2010). Equation (3.13) above illustrates that the wind speed in a continuous probability function ranges between the cut-in and the rated wind speed. Equation (3.14) determines the PDF in the continuous range using the Weibull distribution function in equation (3.9).

$$f_{P_w}(p_w) = \frac{kh_w v_C}{P_R c}\left[\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right]^{k-1} exp\left\{-\left[\left(\frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right]\right., v_C \le v \le v_R \qquad (3.14)$$

where the intermediary parameter $h_w$ is known by: $h_w = \frac{v_R - v_C}{v_C}$.

The wind speed is between the rated and furling wind speed for the discrete probability function, and has a constant zero output power, i.e., below the cut-in and cut-out wind speed as shown in equation (3.13) above. The mathematical expression of the two events $P_w = 0$ and $P_w = P_R$ are illustrated in equations (3.15) and (3.16) separately(Ghasemi et al., 2016).

$$Pr(P_w = 0) = Pr(V < v_C) + Pr(V \ge v_F)$$

$$= 1 - exp\left[-\left(\frac{v_C}{c}\right)^k\right] + 1 - Pr(V < v_F = 1 - exp[-(v_C/c)^k] + 1 - F(V) \qquad (3.15)$$

$$= 1 - exp[-(v_C/c)^k] + 1 - [1 - exp[-(v_F/c)^k]]$$

$$= 1 - exp[-(v_C/c)^k] + exp[-(v_F/c)^k]$$

$$
\begin{aligned}
&Pr(P_w = P_R) = Pr(v_R \le V \le v_F) \\
&= Pr(V \le v_F) - Pr(V \ge v_R) \\
&= 1 - exp[-(v_F/c)^k] - [1 - exp[-(v_R/c)^k]] \\
&= exp[-(v_R/c)^k] - exp[-(v_F/c)^k]
\end{aligned}
\tag{3.16}
$$

The Cumulative Distribution Function (CDF) of wind power is represented by both continuous and discrete probabilities accordingly. The succeeding equation (3.17) characterises the CDF of the output power stochastic variable $P_w$ (Qu et al., 2016; Xian Liu & Xu, 2010).

$$
\begin{cases}
0 & \mathrm{P} < 0 \\
1 & \mathrm{P} \ge P_R \\
1\text{-}\, exp\left\{-\left[\left(1 + \frac{\left(1+\frac{hw}{P_R}\right)v_C}{c}\right)^k\right] + exp\left\{-\left[\frac{v_F}{c}\right]^k\right\}; 0 \le P < P_R\right.
\end{cases}
\tag{3.17}
$$

The system operator may overestimate or underestimate exact availability of wind power output from wind generating systems since the current wind speed is a random variable at all times. Underestimation and overestimation must be included in consideration in the cost function due to wind power output that is typically nonlinear. Equation (3.18) provides a mathematical representation of these two concepts that were previously discussed in equation (3.3). The overestimation and underestimation of wind power generators is specified by equations (3.18)-(3.19) and (3.20)-(3.21), respectively (A. Ghasemi et al., 2016). The overestimation is given by equation (3.18)

$$
C_{r,w,i}(P_{w_i} - P_{W_{i,av}}) = k_{r,i}(P_{w_i} - P_{W_{i,av}}) = k_{r,i}\int_0^{w_i}(P_{w_i} - P_W)\, f_{P_W}(w)dw \tag{3.18}
$$

where the reserve cost coefficient for the $i^{th}$wind power generator, $k_{r,i}$ ,$P_{w_i}$ is the forecasted wind power, $P_{W_{i,av}}$ is the available/generated stochastic wind power and $f_{P_W}(w)dw$ is the probability distribution function for a stochastic variable $P_W$. Now, by utilising the link established in equation (3.17). Equation (3.19) illustrates the further modification of the overestimation equation (3.18).

$$
\begin{aligned}
E(Y_{oe,j,t}) = w_{j,t}&\left[1 - exp\left(-\frac{v_{C\,j,t}}{c_{j,t}}\right)^{k_{j,t}} + exp\left(-\frac{v_{F\,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] + \\
&+\left(\frac{P_{R,j,t}v_{C,j,t}}{v_{R,j,t}-v_{C,j,t}} + w_{j,t}\right)\left[exp\left(-\frac{v_{C\,j,t}}{c_{j,t}}\right)^{k_{j,t}} - exp\left(-\frac{v_{1\,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] + \\
+\left(\frac{P_{R,j,t}c_{j,t}}{v_{R,j,t}-v_{C,j,t}}\right)&\left[\Gamma\left(1 + \frac{1}{k_{j,t}}, \left(\frac{v_{1,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right) - \Gamma\left(1 + \frac{1}{k_{j,t}}, \left(\frac{v_{R,j,t}}{c^{k,j,t}}\right)^{k_{j,t}}\right)\right]
\end{aligned}
\tag{3.19}
$$

Underestimation analysis is performed by using the same approach as overestimation evaluation. Equation (3.20) represents the underestimation of wind power.

$$
C_{p,w,i}(P_{W_{i,av}} - P_{w_i}) = k_{p,i}(P_{W_{i,av}} - P_{w_i}) = k_{p,i}\int_0^{w_i}(P_W - P_{w_i})\, f_{P_W}(w)dw \tag{3.20}
$$

Where the reserve cost coefficient for the $i^{th}$wind power generator, $k_{p,i}$ . Equation (3.21) illustrates a further modification of the underestimation equation (3.20).

$$E(Y_{ue,j,t}) = (P_{R,j,t} - w_{j,t})\left[exp\left(-\frac{v_{R,j,t}}{c_{j,t}}\right)^{k_{j,t}} - exp\left(-\frac{v_{F,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] +$$
$$+\left(\frac{P_{R,j,t}v_{C,j,t}}{v_{R,j,t}-v_{C,j,t}} + w_{j,t}\right)\left[exp\left(-\frac{v_{R,j,t}}{c_{j,t}}\right)^{k_{j,t}} - exp\left(-\frac{v_{1,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right] + \tag{3.21}$$
$$+\left(\frac{P_{R,j,t}c_{j,t}}{v_{R,j,t}-v_{C,j,t}}\right)\left[\Gamma\left(1+\frac{1}{k_{j,t}},\left(\frac{v_{1,j,t}}{c_{j,t}}\right)^{k_{j,t}}\right) - \Gamma\left(1+\frac{1}{k_{j,t}},\left(\frac{v_{R,j,t}}{c^{k,j,t}}\right)^{k_{j,t}}\right)\right]$$

### 3.3 Optimisation of a single area dispatch problem with multi-criteria

WTEED is a multi-criteria optimisation problem with two different goals. One is to reduce the wind and thermal economic dispatch problems overall fuel cost. To reduce the power systems total emission while conforming to the equality and inequality conditions specified in equations (3.4)–(3.8). The max-max price penalty factor stated in equations (3.22) is employed to transform the multi-objective optimisation into a single objective function(Hetzer et al., 2008; Liu & Xu, 2010b). Equations (3.2) and (3.4) represent the fuel cost and emission function, respectively, for the wind-thermal dispatch problem.

$$h_i = \frac{a_i + b_i P_{gimax} + c_i P_{gimax}^2}{\alpha_i + \beta_i P_{gimax} + \gamma_i P_{gimax}^2} \tag{3.22}$$

The price penalty factor is used to transform the bio-objective wind-thermal economic dispatch problem, as illustrated in equation (3.22) above. In this particular case, the maximum fuel cost function divided by the maximum emission function equals the penalty factor. Now, depending on how the optimisation approach is applied, this can alternatively be used as max/min, min/min, or min/max. Equation (3.23) describes how the function of Lagrange, based on a Lagrange multiplier λ, is implemented to solve the optimisation problem.

$$L = F_T + \lambda\left(P_D + P_{Loss} - \sum_{i=1}^{N_G} P_{gi} - \sum_{k=1}^{N_W} P_w\right) \tag{3.23}$$

where

$$F_T = a_i P_{gi}^2 + b_i P_{gi} + c_i + h_i(d_i P_{gi}^2 + e_i P_{gi} + f_i) \tag{3.24}$$

### 3.4 The algorithm applied in solving the single-area WTEED problem

The LaGrange algorithm is one of the best methods for solving optimization problems involving coupled structures. A large-scale linked problem can be converted into a two-stage dual problem with a master problem and subproblems, using Lagrange multipliers to relax global constraints. A suboptimal, viable solution close to the dual optimal point is expected to be recognized as a valid solution to the primal problem based on the sharp bound provided by the Lagrange dual optimum. In a parallel framework, the subproblems can be resolved quickly and easily because they are significantly smaller than the main problem. The dual fitness function is optimized in the master problem stage by updating the Lagrange multipliers. The performance of LR processes is strongly influenced by the methodologies used to initialize and update the Lagrange multipliers.

The LR method is applied to determine the optimal value of the WTEED problem. Dynamic programming's dimensionality challenge can be addressed by temporarily relaxing coupling constraints while focusing on each unit separately (Bhattacharya & Chattopadhyay, 2010). Therefore, reducing fuel expenses and pollution are two issues that the LR strategy can help with. The LaGrange technique solution approach is as follows.

Equation (3.25) is the result of adding the emission and demand limitations to the initial cost function and applying Lagrange multipliers.

$$L = F_T + \lambda\left(P_D + P_{Loss} - \sum_{i=1}^{N_G} P_{gi} - \sum_{k=1}^{N_W} P_w\right) \qquad (3.25)$$

where

$$F_T = a_i P_{gi}^2 + b_i P_{gi} + c_i + h_i(d_i P_{gi}^2 + e_i P_{gi} + f_i) \qquad (3.26)$$

Equation (3.27) is obtained by integrating the loss equation (3.6) into equation (3.25).

$$L = a_i P_{gi}^2 + b_i P_{gi} + c_i + h_i(d_i P_{gi}^2 + e_i P_{gi} + f_i) + \\ +\lambda\begin{pmatrix} P_D + \sum_{m=1}^{N_G+N_W} \quad \sum_{m'=1}^{N_G+N_W} P_m B_{mm'} P_{m'} + \sum_{m=1}^{Ng+Nw} B_{om} P_m \\ +B_{00} - \sum_{i=1}^{N_G} P_{gi} - \sum_{k=1}^{N_W} P_w \end{pmatrix} \qquad (3.27)$$

The condition necessary for optimality for the solution of the problem (3.27) is:

According to $P_{gi,} \frac{\partial L}{\partial P_{gi}} = 0, i = \overline{1, N_g}$ (3.28)

According to $P_{w,} \frac{\partial L}{\partial P_w} = 0, k = \overline{1, N_w}$ (3.29)

According to $\lambda, \frac{\partial L}{\partial \lambda} = 0$ (3.30)

The conditions necessary for optimality for conventional generators, given in equation (3.26), are computed as follows:

$$\frac{\partial L}{\partial P_{gi}} = 2a_i P_{gi} + b_i + h_i\left(2d_i P_{gi} + e_i\right) + \\ +\lambda\left(2\sum_{j=1}^{N_g} B_{ij} P_j + B_{oi} - 1\right) = 0 = i = \overline{1, n} \qquad (3.31)$$

This can be reduced to the following:

$$\frac{\partial L}{\partial P_{gi}} = (2a_i + h_i 2d_i)P_{gi} + 2\lambda B_{ii}P_{gi} + h_i e_i + b_i + \cdots$$

$$\lambda\left(2\sum_{\substack{j=1\\ j\neq i}}^{N_g} B_{ij}P_j + B_{oi} - 1\right) = 0 = i = \overline{1,n} \tag{3.32}$$

By reducing equation (3.32), we have the following equation (3.33)

$$\frac{\partial L}{\partial P_{gi}} = \left(\frac{a_i + h_i d_i}{\lambda} + B_{ii}\right)P_{gi} + \sum_{\substack{j=1\\ j\neq i}}^{n} B_{ij}P_j + \frac{1}{2}\left(\frac{h_i e_i + b_i}{\lambda} + B_{oi} - 1\right) = 0, i = \overline{1,n} \tag{3.33}$$

The matrix representation of equation (3.33) is as follows:

$$\begin{bmatrix} \frac{a_1+h_1d_1}{\lambda} + B_{11} & B_{12} & B_{1n} \\ B_{21} & \frac{a_2+h_2d_2}{\lambda} + B_{22} & B_{2n} \\ B_{n1} & B_{n2} & \frac{a_{1n}+h_nd_n}{\lambda} + B_{nn} \end{bmatrix}\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ \vdots \\ P_n \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \left[1 - \left(\frac{b_1+h_1e_1}{\lambda}\right) - B_{01}\right] \\ \left[1 - \left(\frac{b_1+h_1e_1}{\lambda}\right) - B_{02}\right] \\ \left[1 - \left(\frac{b_1+h_1e_1}{\lambda}\right) - B_{0n}\right] \end{bmatrix} \tag{3.34}$$

The following straightforward matrix form can be used to represent equation (3.34).

$$BP = D \tag{3.35}$$

Equation (3.35) in MATLAB can be used to find the matrix-vector P given the known value of the Lagrange multiplier.

$$P = B/D \tag{3.36}$$

Equation (3.27) provides the necessary conditions for wind generator optimality and can be derived as follows: Equation (3.37) represents the total wind power.

$$\frac{\partial L}{\partial P_w} = \lambda\left(2\sum_{\substack{j=1\\ j\neq i}}^{N_w} B_{ij}P_j + B_{oi} - \frac{\partial}{\partial P_w}(OE + UE + d_iP_w)\right) = 0, i = \overline{1,n} \tag{3.37}$$

Additionally, equation (3.37) can be represented as follows in matrix form:

$$\begin{bmatrix} B_{11} & B_{12} & B_{13} \\ \vdots & \vdots & \vdots \\ B_{n1} & B_{n2} & B_{nn} \end{bmatrix}\begin{bmatrix} P_1 \\ \vdots \\ P_{nw} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \frac{\partial}{\partial P_w}(C_p.UE + C_r.O.E + d_iP_w) - B_{o1} \\ \vdots \\ \frac{\partial}{\partial P_w}(C_p.UE + C_r.O.E + d_iP_w) - B_{onw} \end{bmatrix} \tag{3.38}$$

The conditions necessary for optimality for LaGrange multiplier $\lambda$ in equation (3.27), the coefficients are found as follows:

$$\frac{\partial L}{\partial \lambda} = \left(P_D + P_{Loss} - \sum_{i=1}^{N_G} P_{gi} - \sum_{k=1}^{N_W} P_w\right) = 0 = \Delta\lambda \tag{3.39}$$

The gradient of the functional cap L according to lambda is not represented by the Lagrange multiplier's necessary requirements for optimality. The gradients' optimality conditions must be met. Because equation (3.39) cannot be solved analytically, the gradient method must be implemented as follows.

$$\lambda^{(k+1)} = \lambda^k + \alpha\Delta\lambda^k, \lambda \neq 0 \tag{3.40}$$

where the steps of the gradient procedure, $\alpha$ and $\Delta\lambda^k$is computed using equation (3.40). The gradient approach always begins with an initial guess of a Lagrange multiplier. Upon the value of $\frac{\partial L}{\partial \lambda} = 0$,The solution value for the Lagrange variable is obtained. It determines the energy that the generators must produce as a solution to equations (3.34) and (3.38). Every gradient method step's resultant solution must fit the constraint's domain in the manner that is stated below:

$$P_i^k = \begin{pmatrix} P_{i,min}^k if & P_i^k < & P_{i,min} \\ P_i^k \ if & P_{i,min}^k \leq & P_i^k < P_{i,max} \\ P_{i,min} if & P_i^k > & P_{i,max} \end{pmatrix} \tag{3.41}$$

Equation (3.42) below provides the condition for terminating the gradients and iterations.

$$\Delta\lambda^k \leq \varepsilon, \text{ and } k = iter\ Max \tag{3.42}$$

where, $\varepsilon > 0$ is a smaller number and the maximum number of iterations, $iter\ Max$. The algorithm of the approach is given as follows:

- The Lagrange multiplier's initial value $\lambda^0$ and the optimality conditions value $\varepsilon$ are given.
- Matrix $B^0$ and $D^0$ are formed in equation (3.34)
- Equation (3.36) is resolved and $P^0 = B^0/D^0$is determined
- Wind power is calculated using equations (3.18–3.21). In MATLAB, the trapezoidal method is used to calculate wind power.
- The total power produced by combined wind and thermal generators is calculated by adding the optimum power calculation from equations (3.36) and (3.18-3.21).
- The attained vector of $P^0$ is fitted to the limits equation (3.41).
- $\Delta\lambda^0$is computed with Equation (3.39), where $P^0$ is added.
- The condition (3.41) is examined. If it is satisfied, the process stops. But if it is not, the better value of $\lambda \rightarrow \lambda^1$is computed with Equation (3.40).
- Computation of the better $P_i \rightarrow {P_i}^1$is done following step 5, and the procedure repeats itself until the conditions of Equation (3.42) are satisfied.

The optimal solution is utilised to compute the total cost of generation and emission pollutants employing equations (3.2) and (3.4). Figure 3.2 below shows the algorithm's flow chart.

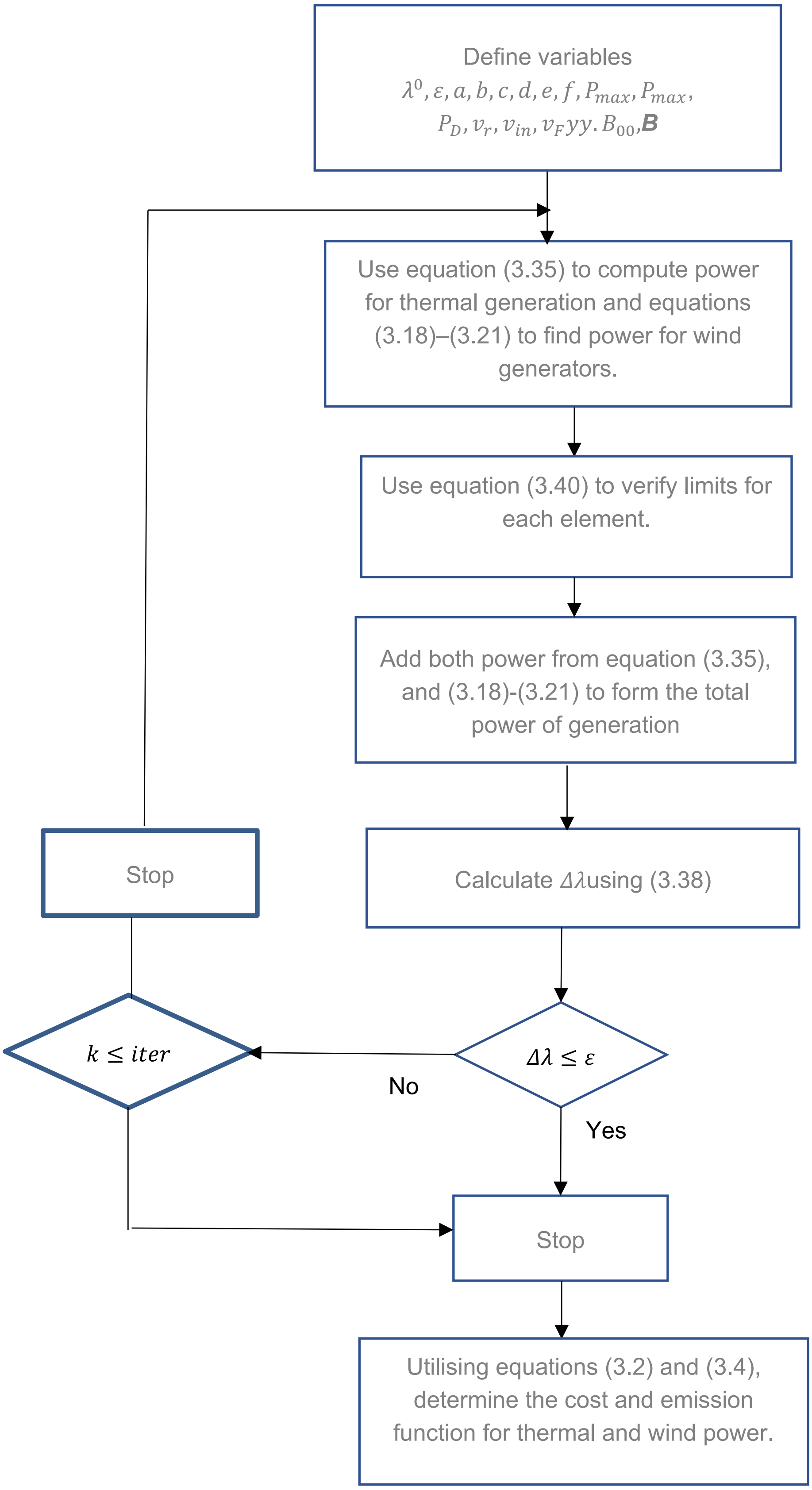



**Figure 3.2:** **Flow diagram of the bi-criteria dispatch problem algorithm solution.**

### 3.5 Description of the Test Systems

The power system comprises six conventional generators and a wind turbine cluster. A line diagram of a basic interconnected power system is illustrated in Figure 3.3. Then, it can be estimated that the six thermal units produce 270 MW and the wind turbines generate 125 MW. On the other hand, interconnected transmission lines enable power to be delivered from one location to another. The simplified power system model in Figure 3.3 highlights the research's focus on the wind-thermal economic dispatch problem. During the day, the power consumption fluctuates between 125 and 300 MW. Tables 3.1-3.3 provide all the necessary information for the WTEED problem. Table 3.4 presents the solution to the wind-thermal economic dispatch problem.

**Table 3.1:** **IEEE 30 Bus System data for 6 thermal systems.**

| Generator Number | Generator Limits [MW] | | Fuel cost coefficients | | | Emission coefficients | | |
|---|---|---|---|---|---|---|---|---|
| | $P_{min}$ | $P_{max}$ | $a_i$[\$/MW$^2$h] | $b_i$[\$/MWh] | $c_i$[\$/h] | $d_i$[\$/kg$^2$h] | $e_i$[\$/kgh] | $f_i$[\$/h] |
| 1 | 50 | 200 | 0.00375 | 2.00 | 0.00 | 0.0126 | -0.90 | 22.983 |
| 2 | 20 | 80 | 0.01750 | 1.70 | 0.00 | 0.0200 | -0.10 | 25.313 |
| 3 | 15 | 50 | 0.06250 | 1.00 | 0.00 | 0.0270 | -0.01 | 25.505 |
| 4 | 10 | 35 | 0.00834 | 3.25 | 0.00 | 0.0291 | -0.005 | 24.900 |
| 5 | 10 | 30 | 0.02500 | 3.00 | 0.00 | 0.0290 | -0.004 | 24.700 |
| 6 | 12 | 40 | 0.02500 | 3.00 | 0.00 | 0.0271 | -0.0055 | 25.300 |

**Table 3.2:** **Transmission line coefficients**

| $B_{0i}$ | | | B | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0.000003 | 0.000021 | -0.00056 | 0.000218 | 0.000103 | 0.000009 | 0.000010 | 0.000002 | 0.000027 |
| 0.000034 | 0.000015 | 0.000078 | 0.000103 | 0.000181 | 0.000004 | 0.000015 | 0.000002 | 0.000030 |
| $B_{00}$ = 0.000014 | | | 0.000009 | 0.000004 | 0.000417 | -0.000131 | -0.000153 | 0.000107 |
| | | | 0.000010 | 0.000015 | 0.000131 | 0.000221 | 0.000094 | 0.000050 |
| | | | 0.000002 | 0.000002 | -0.000153 | -0.000094 | -0.000243 | -0.000001 |
| | | | 0.000027 | 0.000030 | 0.000107 | 0.000050 | -0.000001 | 0.0003458 |

**Table 3.3:** **Wind-speed data**

| $v_{in}$ Cut-in wind-speed | $v_r$ Rated wind-speed | $v_o$ Cut-out wind speed | $w_r$ Rated Power | $C_{pw}$ Underestimation | $C_{rw}$ Overestimation | $d_w$ direct cost |
|---|---|---|---|---|---|---|
| 5 | 15 | 45. | 0.8 | 100 | 310 | 0 |

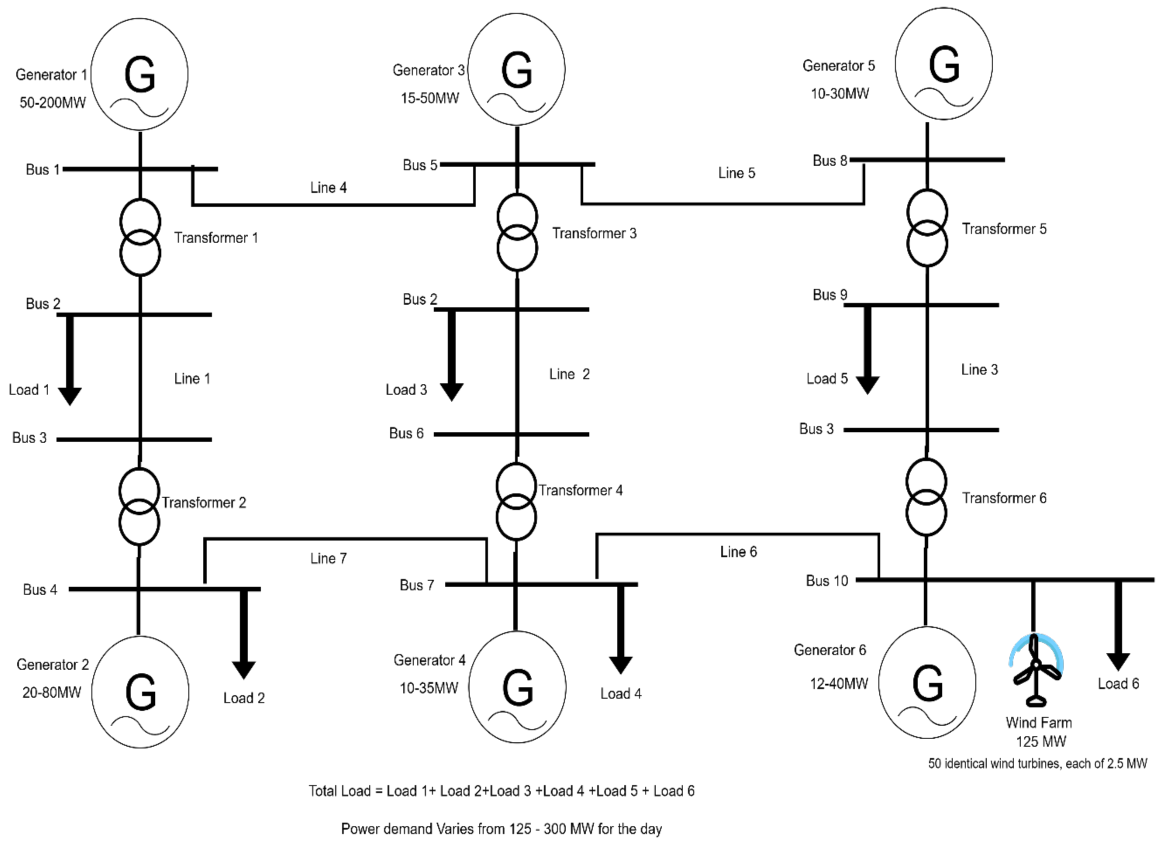



**Figure 3.3:** **Wind-thermal economic dispatch power system model with six generators.**

### 3.5.1 Test System 1: IEEE30 bus system with 6 units

A cluster consists of turbines situated in a geographic area, represented by a group of identical turbines in a large wind farm. The fuel cost coefficients are based on the ones reported in(Damodaran & Kumar, 2014; Krishnamurthy & Tzoneva, 2012), and, for wind power, ( Liu, 2010; Mishra & Shaik, 2024a; Xian Liu & Xu, 2010). The WTEED problem is solved for a load demand of 283.4 MW, using a wind farm comprising 50 identical 2.5-MW wind turbines, for a total installed wind farm capacity of 125 MW. (Kumari & Pal, 2022). Different power demand values are simulated to test the system's behavior under varying demand. The solution to the single-area WTEED problem is shown in Table 3.4. The software is provided in Appendix A1, and the MATLAB script is named WTEED_case6units.m.

**Table 3.4:** **The WTEED using LMM with different power demand for a 6-unit system**

| PD [MW] | 125 | 150 | 175 | 200 | 225 | 250 | 283.4 |
|---|---|---|---|---|---|---|---|
| P1 [MW] | 5 | 5 | 6.65 | 9.54 | 12.44 | 15.35 | 19.24 |
| P2 [MW] | 28.29 | 30.37 | 32.32 | 34.17 | 36.03 | 37.89 | 40.37 |
| P3 [MW] | 14.5 | 20.87 | 26.85 | 32.54 | 38.24 | 43.96 | 51.63 |
| P4 [MW] | 50.68 | 57.51 | 63.91 | 69.98 | 76.06 | 82.15 | 90.31 |
| P5 [MW] | 13.72 | 19.98 | 25.85 | 31.42 | 37.01 | 42.61 | 50.11 |
| P6 [MW] | 13.15 | 16.9 | 20.4 | 23.71 | 27.02 | 30.32 | 34.72 |
| P7 [MW] | 118.28 | 118.28 | 118.28 | 118.28 | 118.28 | 118.28 | 118.28 |
| PL [MW] | 1.32 | 1.63 | 1.97 | 2.36 | 2.79 | 3.26 | 3.96 |
| $F_C$ [USD/h] | 25,391 | 29,234 | 33,164 | 37,160 | 41,167 | 45,183 | 50,562.51 |
| $E_T$ [kg/h] | 22.26 | 21.62 | 21.05 | 20.55 | 20.17 | 19.89 | 19.69 |
| CEED [kg/h] | 28,361 | 33,066 | 37,966 | 43,039 | 48,265 | 53,644 | 61,070 |

Table 3.4 shows the real power of the thermal (P1–P7) units, wind (P7) generators, transmission losses ($P_L$) in [MW], fuel cost ($F_C$) for thermal generators) in MW, total emissions ($E_T$) in kg/h, and CEED in kg/h.

#### 3.5.1.1 WTEED problem discussion and results for the six-unit system

This system uses 50 available wind turbines to provide 118.28 MW to solve the power dispatch problem. The Lagrange multiplier algorithm employs the following parameters. The tolerance value epsilon is 0.01, the tolerance value of delta-lambda epsilon is 0.1, the initial guess value of lambda is 200, and the total number of iterations m is 100.

The proposed method for a six-unit system is applied and compared with different optimisation algorithms found in the literature, including the Genetic Algorithm (GA) (Zhang et al., 2019), the Particle Swarm Optimization (PSO), the Gravitational Search Algorithm (GSA), and the hybrid optimization algorithm Gravitational acceleration Enhanced Particle Swarm Optimization Algorithm (GAEPSO) (Jiang et al., 2015). Table 3.5 presents the comparison of simulation results.

**Table 3.5:** **LMM-based WTEED in comparison to other optimisation techniques. PD = 283.4 [MW]**

| Algorithm | PSO [MW] | GSA [MW] | GAEPSO [MW] | GA [MW] | GPSOA Without Wind | Proposed LMM [MW] |
|---|---|---|---|---|---|---|
| P1 [MW] | 2 | 14.45 | 20.75 | 36.62 | 11.29 | 19.24 |
| P2 [MW] | 30.06 | 53.46 | 30.09 | 42.58 | 29.95 | 40.37 |
| P3 [MW] | 28.24 | 23.87 | 34.63 | 30.31 | 52.66 | 51.63 |
| P4 [MW] | 31.29 | 6 | 35.06 | 31.36 | 101.69 | 90.31 |
| P5 [MW] | 40.32 | 25.48 | 34.53 | 53.44 | 52.68 | 50.11 |
| P6 [MW] | 41.61 | 22.67 | 32.93 | 43.52 | 35.95 | 34.72 |
| Pw [MW] | 110.2 | 137.71 | 95.47 | 45.57 | -- | 118.28 |
| $P_L$ [MW] | --- | --- | --- | --- | --- | 3.96 |
| $F_C$ [USD/h] | 62,497 | 58,846 | 57,219 | 66,582 | 60,029 | 50,562.51 |
| % Deviation FC [$/h] Compared to Developed LMM | 5.27 | 3.78 | 3.08 | 6.83 | 4.10 | 4.27 |
| $E_T$ [kg/h] | 20.43 | 21.33 | 20.49 | 19.72 | 22.21 | 19.69 |
| % Deviation $E_T$ [kg/h] Compared to Developed LMM | 0.922 | 1.99 | 0.99 | 0.038 | 3.007 | 3.007 |

Compared with the LMM algorithm, the fuel cost values for the PSO, GSA, GAEPSO, GA, and GPSOA algorithms are 5.27, 3.78, 3.08, 6.83, and 4.27, respectively. Compared to other optimisation algorithms, the LLM method reduces fuel usage by 3-6% when wind power is incorporated in the economic emission dispatch. Additionally, the resultant NOx emissions are 0.922, 1.99, 0.99, 0.038, and 3.01, respectively, compared to the proposed algorithm. The effectiveness of the Lagrange algorithm in reducing emissions when wind power is added to conventional generation decreased by 1–3%. The optimization algorithm required only 84 iterations to find the optimal solution to the WTEED problem. The calculated value of lambda is 214.9919, compared to the initial guess of 200. Figures 3.4 and 3.5 below represent the fuel cost and emissions.

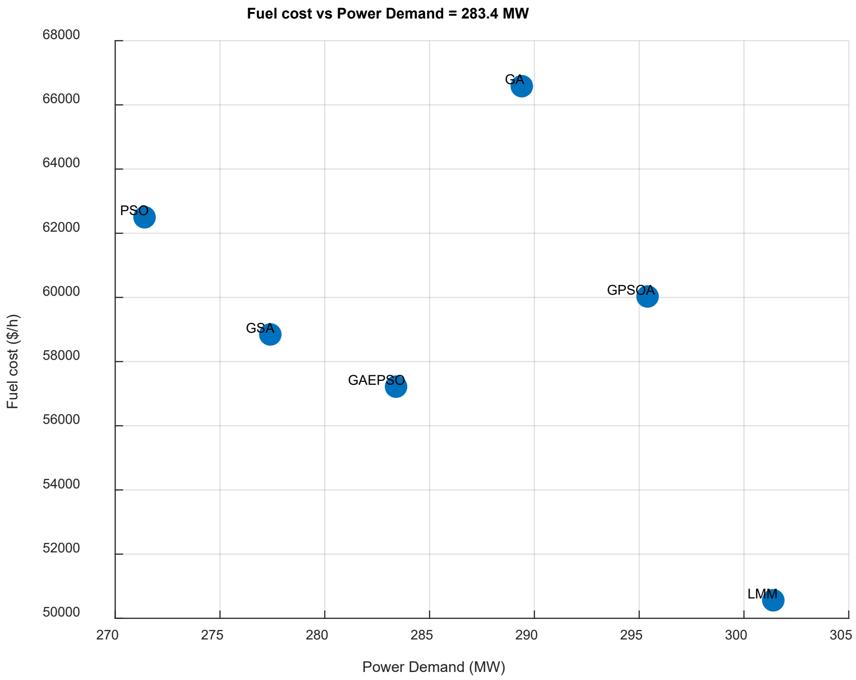



**Figure 3.4:** Fuel cost and emissions for various algorithms applied to a 6-unit system

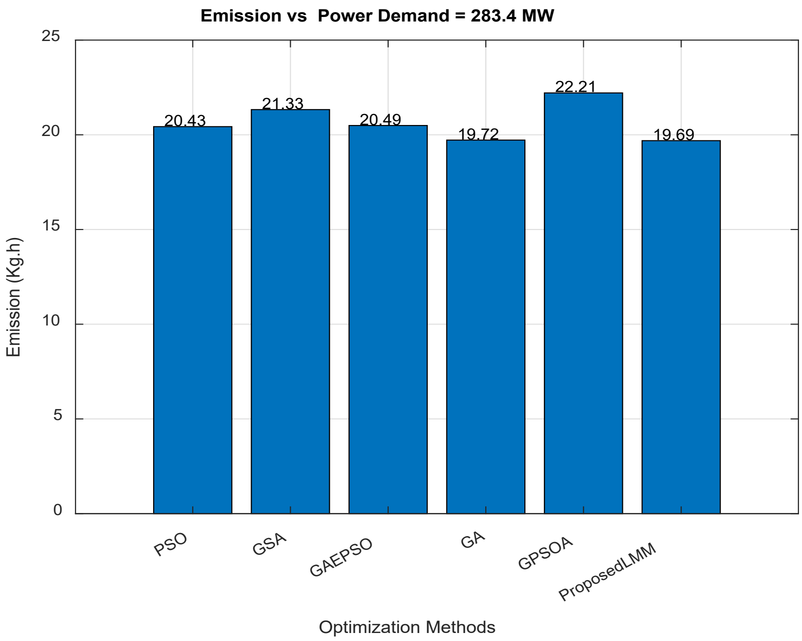



**Figure 3.5:** Comparison of the Fuel cost and emission analysis for different optimization algorithms applied to a 6-unit system.

### 3.5.2 Test System 2: IEEE30 bus system with 10-units.

This study of the WTEED power system problem includes 10 thermal generators and 1 wind turbine cluster. The cluster consists of turbines situated in a specific geographic area, represented by a group of identical turbines in a large wind farm. The fuel cost, emission, and loss coefficients are based on those reported in (Basu, 2011a). The wind parameters are adopted from (Liu, 2010; Mishra & Shaik, 2024a; Xian Liu & Xu, 2010), as in Test System I above. The study focuses on 2000 MW power demand.(Secui et al., 2024a). Various power demand values are used to test the system's response to changes in power demand. The solution of the WTEED problem is shown in Table 3.6. The software is provided in Appendix A2, and the MATLAB script file is named WTEED_case10units. m.

**Table 3.6:** **Various power demands applied to a 10-unit generator for LMM WTEED problem**

| PD [MW] | 650 | 800 | 1000 | 1200 | 1500 | 2000 |
|---|---|---|---|---|---|---|
| P1 [MW] | 16.9077 | 43.2946 | 55.0000 | 55.0000 | 55.0000 | 55.0000 |
| P2 [MW] | 20.4349 | 45.0263 | 59.3561 | 69.8398 | 80.0000 | 80.0000 |
| P3 [MW] | 47.0000 | 47.0000 | 60.9549 | 71.9979 | 87.5874 | 111.9414 |
| P4 [MW] | 20.0000 | 44.1126 | 59.4937 | 70.7465 | 86.6321 | 111.4487 |
| P5 [MW] | 50.0000 | 50.0000 | 50.0000 | 50.0000 | 67.6960 | 111.5112 |
| P6 [MW] | 70.0000 | 70.0000 | 70.0000 | 70.0000 | 78.8096 | 134.2497 |
| P7 [MW] | 60.0000 | 62.8499 | 106.5857 | 138.5827 | 183.7531 | 254.3184 |
| P8 [MW] | 70.0000 | 70.0000 | 114.6430 | 152.1590 | 205.1205 | 287.8573 |
| P9 [MW] | 135.0000 | 135.0000 | 175.7180 | 229.6450 | 305.7741 | 424.7033 |
| P10 [MW] | 150.0000 | 150.0000 | 173.2323 | 228.4474 | 306.3949 | 428.1648 |
| P11 [MW] | 24.028 | 100.116 | 100.116 | 100.116 | 100.116 | 100.116 |
| PL [MW] | 13.3715 | 17.4007 | 25.1010 | 36.5356 | 56.8851 | 99.3099 |
| $F_C$ $/h] | 37,639.687359 | 41,146.509 | 50,742.1419 | 60,746.3 | 76,945.943152 | 107,422.761421 |
| $E_T$ [kg/h] | 1399.058217 | 1333.454716 | 1501.85983 | 1803.17 | 2440.652768 | 3910.593584 |
| CEED [kg/h] | 63,739.681832 | 66,384.319239 | 80,126.5769 | 97,793.9 | 131,033.780343 | 203,261.325452 |

Table 3.6 shows the real power of the thermal (P1–P11) units, wind (P11) generators, and transmission losses $(\boldsymbol{P_L})$ in MW, fuel cost $\boldsymbol{F_C}$ for thermal generators, total emissions $(\boldsymbol{E_T})$ in kg/h, and CEED in kg/h.

#### 3.5.2.1 WTEED problem results and discussion for Ten-Unit System

The Lagrange method is implemented to solve the economic emission dispatch problem with wind power, using 10 conventional units and 50 available wind turbines on a wind farm consisting of 50 identical wind turbines. The following parameters are used for the Lagrange multiplier method. The initial value of lambda is 200, the total number of iterations m is 3000, the tolerance value (epsilon) is 0.01, and the tolerance value of delta-lambda epsilon is 0.1. The Lagrange multiplier method is compared with the Salp Swarm Algorithm (SSA), Grey Wolf Optimization (GWO), Moth–Flame Optimization (MFO), Whale Optimization Algorithm (WOA), Search and Rescue Algorithm (SAR), and Genetic Algorithm (GA) (Mishra & Shaik, 2024a). The optimization algorithm took 1013 iterations to find the optimal solution of the WTEED problem. The optimized lambda value is 151.2992, compared to the initial guess of 200. The proposed algorithm has a lower fuel cost and is observed to be superior to all other optimization algorithms, as shown in Table 3.7.

**Table 3.7:** **LMM-based WTEED in comparison to other optimisation techniques. PD = 2000[MW]**

| Algorithm | MFO | WOA | SAR | GA | Proposed LMM |
|---|---|---|---|---|---|
| P1 [MW] | 37.16 | 46.96 | 53.49 | 34.17 | 55.0000 |
| P2 [MW] | 25.84 | 68.61 | 77.10 | 63.49 | 80.0000 |
| P3 [MW] | 120 | 69.24 | 64.41 | 95.75 | 111.9414 |
| P4 [MW] | 93.90 | 129.21 | 83.05 | 121.67 | 111.4487 |
| P5 [MW] | 160 | 50 | 158.73 | 145.83 | 111.5112 |
| P6 [MW] | 240 | 226 | 239.50 | 229.86 | 134.2497 |
| P7 [MW] | 244.61 | 298.21 | 275.88 | 281.21 | 254.3184 |
| P8 [MW] | 286.25 | 337.93 | 332.28 | 307.35 | 287.8573 |
| P9 [MW] | 470 | 390.40 | 409.12 | 391.63 | 424.7033 |
| P10 [MW] | 404.92 | 468 | 388.56 | 410.56 | 428.1648 |
| P11 [MW] | 30 | 30 | 30 | 30 | 100.116 |
| Fc [USD] | 116,690.8 | 114,678.8 | 114,106.19 | 115,802.24 | 107,422.761421 |
| % Deviation $F_C$ [$/h] Compared to Developed LMM | 2.07 | 1.63 | 1.508477 | 1.876913 | 1.876 |
| $F_{CT}$ [$/h] | 239,158.6 | 244,987.2 | 226,506.78 | 231,903.60 | 110,572.517314 |
| PL [MW] | 83.70 | 84.44 | 82.81 | 83.9 | 99.310 |
| $E_T$ [kg/h] | 4080 | 4220 | 4103.3 | 4390 | 3910.593584 |
| % Deviation $E_T$ [kg/h] Compared to Developed LMM | 1.060036 | 1.90273 | 1.202327 | 2.897008 | 2.887784 |
| CEED[[kg/h]] | 239,158.6 | 244,987.2 | 226,506.78 | 231,903.60 | 203,261.325452 |

Compared with the LMM algorithm, the fuel cost obtained with the MFO, WOA, SAR, and GA algorithms are 2.07%, 1.63%, 1.5%, and 1.89%, respectively. The proposed method outperforms other algorithms by 1-2% in solving the wind-thermal economic emission dispatch problem. Compared with the developed LM algorithm, the resulting NOx emissions are 1.06%, 1.9%, 1.2%, and 2.89%, respectively. Compared with other optimisation techniques, the LM approach has successfully reduced pollution levels by 1-2.89%.

Figures 3.6 and 3.7 below provide a graphic representation of fuel cost and emissions.

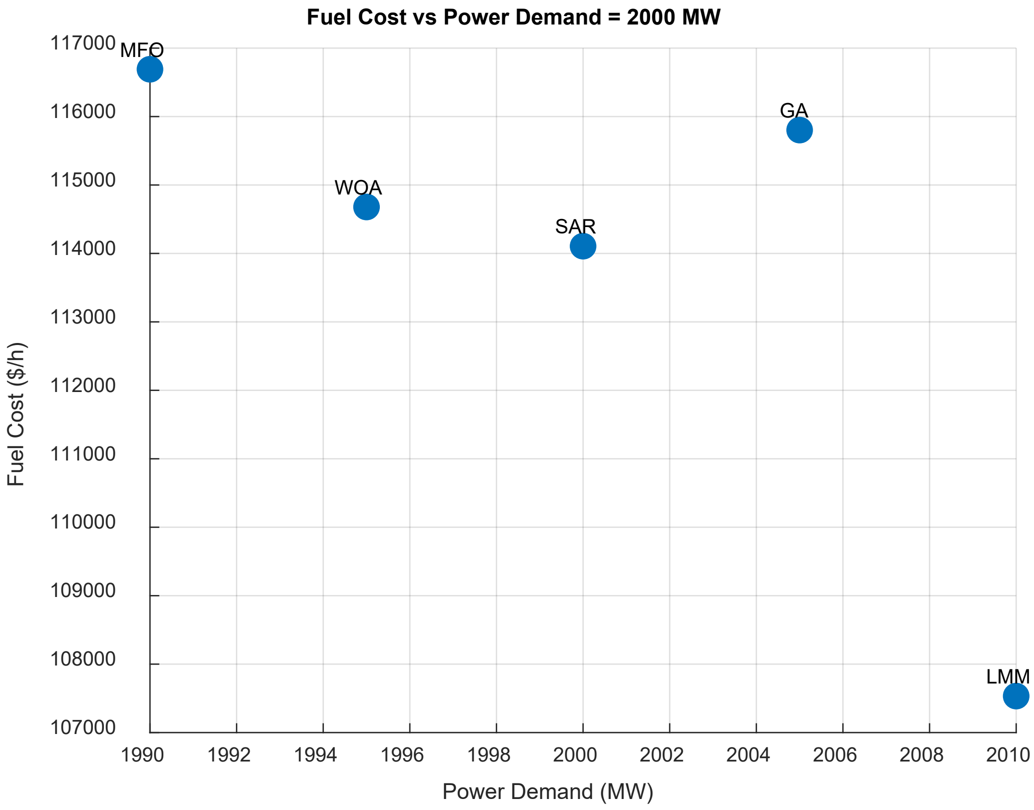



**Figure 3.6:** Values of fuel cost and power demand for various algorithms employing a 10-unit system in comparison to the developed algorithm

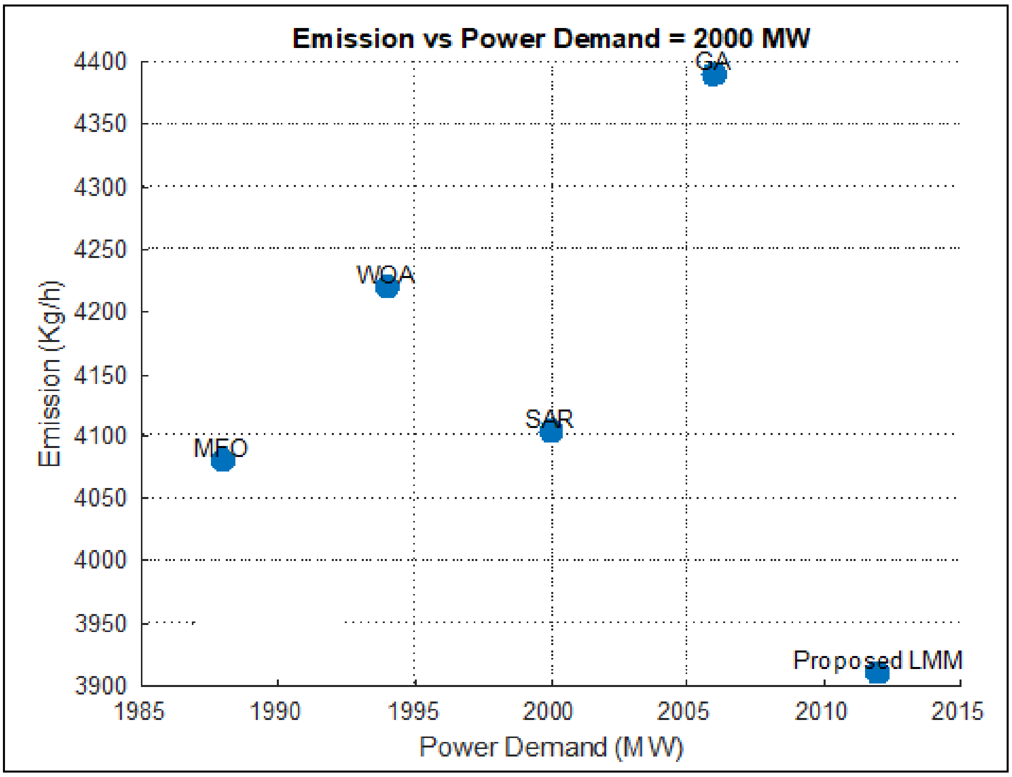



**Figure 3.7**: Values of emission and power demand for various algorithms employing a 10-unit system in comparison to the developed algorithm

The fuel cost and emission values are further compared to the statistical values of different algorithms, as presented in Table 3.8.

**Table 3.8:** **Developed LMM compared with various algorithms for the ten-unit system** (Secui et al., 2024a)

| Algorithm | Fuel cost ($/h) | Fuel cost (%) Deviation | Emission (kg/h) | Emission (%) Deviation |
|---|---|---|---|---|
| DE | 111,500.000 | 0.931205 | 3923.40 | 0.081736 |
| QPSO | 119,005.3030 | 2.557665 | 4032.38 | 0.76663 |
| GQPSO | 112,424.7444 | 1.137603 | 4011.9244 | 0.639511 |
| SGO | 111,497.6302 | 0.930674 | 3932.243252 | 0.138022 |
| MSGO | 111,497.6301 | 0.930674 | 3932.243252 | 0.138022 |
| LMM | 107,422.761421 | 1.297564 | 3910.593584 | 0.352784 |

In Table 3.8, fuel cost and emissions for different algorithms used to solve the thermal economic problem are compared with those of the proposed LMM. It is found that the LMM is 1.29% and 0.353% better at minimizing fuel and emissions.

### 3.5.3 Test System 3: IEEE30 bus system with 40-units

The test system consists of a 40-unit thermal power generator with a power demand of 10,500 MW, accounting for power losses without valve-point loading. The simulation data for power limits, fuel cost coefficients, emission coefficients, and B-loss coefficients used in this test system are identical to those used in(Secui et al., 2024a), also presented in Tables 3.9 and 3.10(a-b) respectively.

**Table 3.9:** IEEE 30 Bus System data for 40 thermal systems

| Generating Unit [MW] | Generator Limits | | Fuel cost coefficients | | | Emission coefficients | | |
|---|---|---|---|---|---|---|---|---|
| | $P_{min}$ | $P_{max}$ | $a_i$ [$/(MWh)$^2$] | $b_i$ [$/MWh] | $c_i$ [$/h] | $\alpha_i$ [t/(MWh)$^2$] | $\beta_i$ [t/MWh] | $\gamma_i$ [t/h] |
| 1 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 2 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 3 | 60 | 120 | 0.02028 | 7.07 | 309.54 | 0.0762 | −2.36 | 100 |
| 4 | 80 | 190 | 0.00942 | 8.18 | 369.03 | 0.054 | −3.14 | 120 |
| 5 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 6 | 68 | 140 | 0.01142 | 8.05 | 222.33 | 0.0854 | −3.08 | 80 |
| 7 | 110 | 300 | 0.00357 | 8.03 | 287.71 | 0.0242 | −3.06 | 100 |
| 8 | 135 | 300 | 0.00492 | 6.99 | 391.98 | 0.031 | −2.32 | 130 |
| 9 | 135 | 300 | 0.00573 | 6.6 | 455.76 | 0.0335 | −2.11 | 150 |
| 10 | 130 | 300 | 0.00605 | 12.9 | 722.82 | 0.425 | −4.34 | 280 |
| 11 | 94 | 375 | 0.00515 | 12.9 | 635.2 | 0.0322 | −4.34 | 220 |
| 12 | 94 | 375 | 0.00569 | 12.8 | 654.69 | 0.0338 | −4.28 | 225 |
| 13 | 125 | 500 | 0.00421 | 12.5 | 913.4 | 0.0296 | −4.18 | 300 |
| 14 | 125 | 500 | 0.00752 | 8.84 | 1760.4 | 0.0512 | −3.34 | 520 |
| 15 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 16 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 17 | 220 | 500 | 0.00313 | 7.97 | 647.85 | 0.0151 | −2.68 | 220 |
| 18 | 220 | 500 | 0.00313 | 7.95 | 649.69 | 0.0151 | −2.66 | 222 |
| 19 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |
| 20 | 242 | 550 | 0.00313 | 7.97 | 647.81 | 0.0151 | −2.68 | 220 |
| 21 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 290 |
| 22 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 285 |
| 23 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 24 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 25 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 26 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 27 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 28 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 29 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 30 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 31 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 32 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 33 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 34 | 90 | 200 | 0.0001 | 8.95 | 107.87 | 0.0012 | −3.48 | 65 |
| 35 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 36 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 37 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 38 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 39 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 40 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |

**Table 3.10(a):** **The B-coefficients B_ij×10^ (-6), B_0i×10^ (-4), and B_00  for 40-unit system**

| Unit | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 2 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 3 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 4 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 5 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 6 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |
| 7 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 8 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 9 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 10 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 11 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 12 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |
| 13 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 14 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 15 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 16 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 17 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 18 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |
| 19 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 20 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 21 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 22 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 23 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 24 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |
| 25 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 26 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 27 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 28 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 29 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 30 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |
| 31 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 32 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 33 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 34 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| 35 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 |
| 36 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 |

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 37 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 |
| 38 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 |
| 39 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 |
| 40 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 |
| $B_{oi}$ | **−3.**908 | **−1.**297 | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | −3.908 | **−1.**297 | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | **−3.**908 | **−1.**297 | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | **−3.**908 | **−1.**297 |

**Table 3.10(b):** **The B-coefficients B_ij×10^ (-6), B_0i×10^ (-4), and B_00 for 40-unit system**

| Unit | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 2 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 3 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 4 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 5 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 6 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |
| 7 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 8 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 10 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 11 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 12 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |
| 13 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 15 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 16 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 17 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 18 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |
| 19 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 20 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 21 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 22 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 23 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 24 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |
| 25 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 26 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 27 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 28 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 29 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 30 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 31 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 32 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 33 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 34 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| 35 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 | 129 | −2 | −5 | −6 | −10 | −6 |
| 36 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 | −2 | 150 | −2 | −1 | −6 | −8 |
| 37 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 | −5 | −2 | 17 | 12 | 7 | −1 |
| 38 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 | −6 | −1 | 12 | 14 | 9 | 1 |
| 39 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 | −10 | −6 | 7 | 9 | 31 | 0 |
| 40 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 | −6 | −8 | −1 | 1 | 0 | 24 |
| **$B_{oi}$** | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | **−3.**908 | **−1.**297 | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | **−3.**908 | **−1.**297 | **7.**047 | **0.**591 | **2.**161 | **−6.**635 | **−3.**908 | **−1.**297 | 7.047 | 0.591 |

The cluster consists of turbines located within a geographic area, represented by a group of identical turbines in a large wind farm, as shown in Test System One. Various power demand values are given to test the system's behavior in response to changes in power demand. The solution of the WTEED problem is shown in Table 3.11 below.

**Table 3.11:** Various power demands applied to a 40-unit generator for the LMM WTEED problem

| PD [MW] | 10,500 | 11,000 | 11,200 | 11,300 |
|---|---|---|---|---|
| P1 [MW] | 114.0000 | 114.0000 | 114.0000 | 114.0000 |
| P2 [MW] | 114.0000 | 114.0000 | 114.0000 | 114.0000 |
| P3 [MW] | 120.0000 | 120.0000 | 120.0000 | 120.0000 |
| P4 [MW] | 190.0000 | 190.0000 | 190.0000 | 190.0000 |
| P5 [MW] | 97.0000 | 97.0000 | 97.0000 | 97.0000 |
| P6 [MW] | 140.0000 | 140.0000 | 140.0000 | 140.0000 |
| P7 [MW] | 300.0000 | 300.0000 | 300.0000 | 300.0000 |
| P8 [MW] | 300.0000 | 300.0000 | 300.0000 | 300.0000 |
| P9 [MW] | 300.0000 | 300.0000 | 300.0000 | 300.0000 |
| P10 [MW] | 236.1173 | 300.0000 | 300.0000 | 300.0000 |
| P11 [MW] | 279.1346 | 369.8932 | 375.0000 | 375.0000 |
| P12 [MW] | 277.8458 | 369.5691 | 375.0000 | 375.0000 |
| P13 [MW] | 383.4729 | 500.0000 | 500.0000 | 500.0000 |
| P14 [MW] | 422.9295 | 500.0000 | 500.0000 | 500.0000 |
| P15 [MW] | 421.4242 | 500.0000 | 500.0000 | 500.0000 |
| P16 [MW] | 421.4242 | 500.0000 | 500.0000 | 500.0000 |
| P17 [MW] | 500.0000 | 500.0000 | 500.0000 | 500.0000 |
| P18 [MW] | 500.0000 | 500.0000 | 500.0000 | 500.0000 |
| P19 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P20 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P21 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P22 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P23 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P24 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P25 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P26 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.0000 |
| P27 [MW] | 17.9820 | 23.6373 | 101.6061 | 142.1987 |
| P28 [MW] | 17.9820 | 23.6373 | 101.6061 | 142.1987 |
| P29 [MW] | 17.9820 | 23.6373 | 101.6061 | 142.1987 |
| P30 [MW] | 97.0000 | 97.0000 | 97.0000 | 97.0000 |
| P31 [MW] | 190.0000 | 190.0000 | 190.0000 | 190.0000 |
| P32 [MW] | 190.0000 | 190.0000 | 190.0000 | 190.0000 |
| P33 [MW] | 190.0000 | 190.0000 | 190.0000 | 190.0000 |
| P34 [MW] | 90.0000 | 90.0000 | 90.0000 | 90.0000 |
| P35 [MW] | 90.0000 | 90.0000 | 90.0000 | 90.0000 |
| P36 [MW] | 90.0000 | 90.0000 | 90.0000 | 90.0000 |
| P37 [MW] | 110.0000 | 110.0000 | 110.0000 | 110.0000 |
| P38 [MW] | 110.0000 | 110.0000 | 110.0000 | 110.0000 |
| P39 [MW] | 110.0000 | 110.0000 | 110.0000 | 110.0000 |
| P40 [MW] | 550.0000 | 550.0000 | 550.0000 | 550.000 |
| P41 [MW] | 150.386 | 150.386 | 150.386 | 150.386 |
| $F_C$ [USD/h] | 133,540.341 | 143,551.62517 | 159,778.38 | 175,659.5 |
| $P_L$ [MW] | 1038.682 | 1152.8 | 1197.2 | 1218.984 |
| $E_T$ [kg/h] | 105,263.403 | 136,510.06316 | 190,426.05 | 244,979.8 |
| CEED [kg/h] | 254,190.3053 | 283,453.10221 | 317,220.20 | 350,503.7 |

In Table 3.11 above, the real power of the thermal (P1–P40) units, wind (P41) generators, and transmission losses ($\boldsymbol{P_L}$) in MW, fuel cost ($\boldsymbol{F_C}$) for thermal generators, total emissions ($\boldsymbol{E_T}$) in kg/h, and CEED in kg/h.

### 3.5.3.1 WTEED problem results and discussions for Forty-Unit System

This is a fourth-generation unit system used to evaluate the effectiveness of the Lagrange multiplier method relative to other optimization methods for minimizing fuel costs and atmospheric emissions. The following parameters are used for the Lagrange multiplier method. The initial value of lambda is 8.9914, the total number of iterations m is 35,000, the tolerance value (epsilon) is 0.01, and the tolerance value of delta-lambda epsilon is 0.1. The Lagrange multiplier method is compared with different heuristic algorithms, i.e., PSO, GSA, GAEPSO(Jiang et al., 2015) and MSGO(Secui et al., 2024a). The comparison is made based on the best cost and lowest emissions. The summary of the comparison between the Lagrange method and the mentioned heuristic algorithms is given in Table 3.12.

**Table 3.12: The WTEED problem with Lagrange compared with other optimization algorithms for an IEEE 40-unit system for PD = 10,500 MW**

| Method | MGSO Without Wind Power | MSGO | PSO | GSA | GAEPSO | Proposed LMM 150 MW Wind Power | Proposed LMM 1100 MW Wind Power |
|---|---|---|---|---|---|---|---|
| $P_1$ [MW] | 114.000000 | 549.9997828 | 114.0000 | 105.5679 | 106.3768 | 114.0000 | 114.0000 |
| P2 [MW] | 113.999999 | 549.9997818 | 104.0000 | 88.2574 | 113.2637 | 114.0000 | 114.0000 |
| P3 [MW] | 120.000000 | 119.9988116 | 120.0000 | 105.9739 | 108.5784 | 120.0000 | 108.3202 |
| P4 [MW] | 189.999995 | 179.7374211 | 169.3671 | 150.3464 | 188.6623 | 190.0000 | 165.3147 |
| P5 [MW] | 96.999999 | 96.99883835 | 97.0000 | 82.0595 | 77.0086 | 97.0000 | 97.0000 |
| P6 [MW] | 140.000000 | 105.4022758 | 124.2630 | 119.5704 | 132.6636 | 140.0000 | 130.6214 |
| P7 [MW] | 300.000000 | 299.9999742 | 299.6931 | 248.5154 | 288.3156 | 300.0000 | 285.1828 |
| P8 [MW] | 300.000000 | 285.7135868 | 297.9093 | 276.3936 | 235.0264 | 300.0000 | 300.0000 |
| P9 [MW] | 299.999998 | 287.9519754 | 297.2578 | 244.2866 | 285.7760 | 300.0000 | 300.0000 |
| P10 [MW] | 279.599683 | 204.8086989 | 130.0007 | 262.1424 | 264.6783 | 236.1273 | 166.9658 |
| P11 [MW] | 168.799860 | 243.6012566 | 298.4210 | 293.2579 | 312.0426 | 279.1439 | 214.3810 |
| P12 [MW] | 94.0000003 | 243.6003501 | 298.0264 | 264.5149 | 112.8675 | 277.8553 | 212.4040 |
| P13 [MW] | 484.039161 | 394.2851366 | 433.5590 | 432.2395 | 486.8320 | 383.4865 | 289.4015 |
| P14 [MW] | 484.039166 | 394.2799802 | 421.7360 | 391.8179 | 341.2265 | 422.9446 | 318.7482 |
| P15 [MW] | 484.039164 | 394.2841752 | 422.7884 | 422.8119 | 428.9123 | 421.4392 | 317.4792 |
| P16 [MW] | 484.039178 | 484.0292251 | 422.7841 | 414.8810 | 436.8761 | 421.4392 | 317.4792 |
| P17 [MW] | 489.279372 | 489.2692227 | 439.4078 | 428.5659 | 426.2216 | 500.0000 | 462.7621 |
| P18 [MW] | 489.279372 | 399.5209877 | 409.4132 | 422.1613 | 442.5531 | 500.0000 | 464.5057 |
| P19 [MW] | 511.279600 | 506.0067276 | 439.4111 | 440.9423 | 454.6218 | 550.0000 | 506.5800 |
| P20 [MW] | 511.279490 | 421.5364187 | 429.4155 | 435.9092 | 491.0606 | 550.0000 | 506.5812 |
| P21 [MW] | 526.732209 | 433.5530376 | 439.4421 | 451.8724 | 450.2265 | 550.0000 | 550.0000 |
| P22 [MW] | 550.000000 | 514.2353651 | 439.4587 | 436.1765 | 406.2236 | 550.0000 | 550.0000 |
| P23 [MW] | 523.279384 | 433.5329212 | 429.7822 | 437.2970 | 464.5563 | 550.0000 | 550.0000 |
| P24 [MW] | 523.279383 | 433.5268561 | 439.7697 | 442.0350 | 444.5589 | 550.0000 | 550.0000 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| P25 [MW] | 524.239856 | 433.5342317 | 430.1191 | 445.9564 | 455.3451 | 550.0000 | 550.0000 |
| P26 [MW] | 523.815577 | 433.7489257 | 440.1219 | 429.3785 | 468.2267 | 550.0000 | 550.0000 |
| P27 [MW] | 10.000020 | 10.05540017 | 28.9738 | 124.0932 | 103.8549 | 17.9826 | 13.9471 |
| P28 [MW] | 10.000113 | 10.24418652 | 29.0007 | 117.8668 | 146.2286 | 17.9826 | 13.9471 |
| P29 [MW] | 10.000007 | 10.0022597 | 28.9828 | 94.5359 | 148.8421 | 17.9826 | 13.9471 |
| P30 [MW] | 87.800155 | 89.80426078 | 97.0000 | 89.6485 | 96.7263 | 97.0000 | 97.0000 |
| P31 [MW] | 190.000000 | 189.998884 | 172.3348 | 153.1318 | 189.2261 | 190.0000 | 175.9692 |
| P32 [MW] | 190.000000 | 189.9955403 | 172.3327 | 159.0102 | 86.6516 | 190.0000 | 175.9692 |
| P33 [MW] | 190.000000 | 189.9978565 | 172.3262 | 148.7814 | 89.2264 | 190.0000 | 175.9692 |
| P34 [MW] | 200.000000 | 199.9994215 | 200.0000 | 176.4061 | 93.9663 | 90.0000 | 90.0000 |
| P35 [MW] | 199.999999 | 199.9994151 | 200.0000 | 170.0710 | 102.2234 | 90.0000 | 90.0000 |
| P36 [MW] | 164.799870 | 199.998299 | 200.0000 | 181.6662 | 168.2261 | 90.0000 | 90.0000 |
| P37 [MW] | 110.000000 | 110.0000000 | 100.8441 | 96.8108 | 78.2663 | 110.0000 | 110.0000 |
| P38 [MW] | 109.999999 | 109.9997758 | 100.8346 | 94.3094 | 107.8912 | 110.0000 | 110.0000 |
| P39 [MW] | 110.000000 | 109.9962344 | 100.8362 | 82.4816 | 98.0368 | 110.0000 | 110.0000 |
| P40 [MW] | 550.000000 | 509.5675741 | 439.3868 | 456.2560 | 442.4562 | 550.0000 | 506.5800 |
| P41 [MW] | - | 1100 | 66.2736 | 83.099 | 151.9009 | 150.294 | 1100 |
| $F_C$ [$/h] | 136,454.336 | 126,645.1342 | 133,995.061 | 142,675.36 | 158,269.6617 | 133,541.733580 | 121,045.419972 |
| % Deviation $F_C$ [$/h] Compared to Developed LMM | 2.992 | 1.13 | 2.54 | 4.17 | 6.66 | 2.454 | 3.313174 (Average) |
| $P_L$ [MW] | 958.620621 | 962.815076 | - | - | - | 1038.679 | 1063.127 |
| $E_T$ [kg/h] | 347,578.490 | 239,155.7318 | 352.766.822 | 366,190.13 | 156,826.1213 | 105,267.750200 | 75,880.539978 |
| % Deviation $E_T$ [kg/h] Compared to Developed LMM | 32.08 | 25.91371319 | 32.29 | 32.83 | 17.39 | 8.11 | 24.77 (Average) |

Compared with the LMM (1100 MW) algorithm, the fuel costs for the MGSO (without wind), MGSO, PSO, GSA, GAEPSO, and LMM (150 MW) algorithms are 2.99%, 1.33%, 2.54%, 4.17%, 6.66%, and 2.454%, respectively. When solving the WTEED problem, the developed solution outperforms existing algorithms by 1–6%. Compared with the proposed LM method, the resulting NOx emissions are 32.08%, 25.91%, 32.29%, 32.83%, 17.39%, and 8.11%, respectively. Compared with other optimisation techniques, the LM approach has successfully reduced pollution levels by 1–32%.

The fuel cost and emission figures for several algorithms used to solve the thermal economic problem with the developed LMM are compared in Table 3.13. The LMM is found to reduce fuel and emissions by 0.59% and 22.163%, respectively.

**Table 3.13:** **Developed LMM compared with various algorithms for the 40-unit system** (Secui et al., 2024a)

| Algorithm | Fuel cost ($/h) | Fuel cost (%)Deviation | Emission (kg/h) | Emission (%) Deviation |
|---|---|---|---|---|
| PSO | 123,607.9479 | 0.523706 | 193,313.7047 | 21.8119754 |
| DE | 123,804.0394 | 0.56333 | 193,953.9668 | 21.87886 |
| SCA | 125,895.2706 | 0.981987 | 210,484.9674 | 23.50221 |
| SGO | 123,289.7874 | 0.45928 | 193,311.54075 | 21.81175 |
| MGSO | 123,161.8867 | 0.433334 | 193,311.54075 | 21.81175 |
| LMM | 121,045.419972 | 0.592327 | 75,880.5399 | 22.16331 |

The optimal solution to the WTEED problem emerged after 72 iterations of the optimisation procedure. Compared to the initial estimate of 8.9914, the optimised lambda value is 33.9059. In addressing the WTEED problem, the LM algorithm performed better than alternative optimisation algorithms. Additionally, Table 3.13 demonstrates that improving power generation from renewable energy sources can dramatically reduce emissions and fuel cost. Figures 3.8 and 3.9 graphically represent the correlations between fuel cost [USD/h], power demand [MW], and emissions for various optimisation strategies

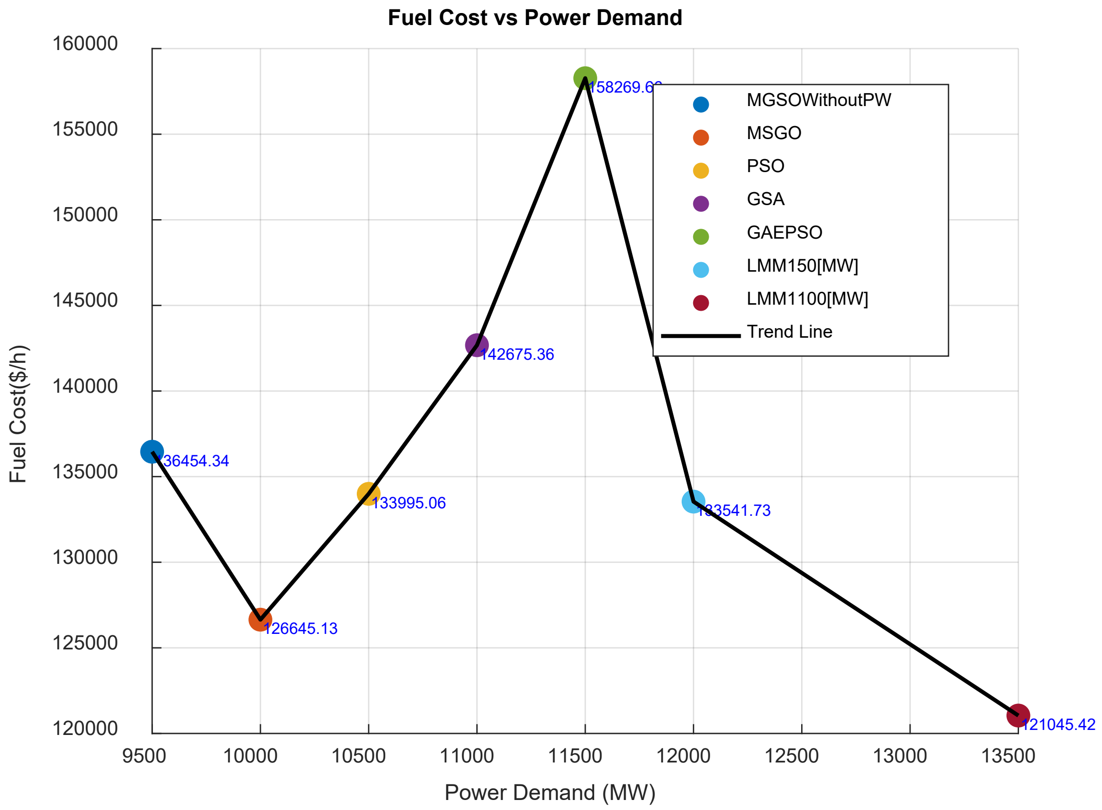



**Figure 3.8:** **Values of Fuel cost and power demand for various algorithms using a 40-unit system in relation to the developed algorithm**

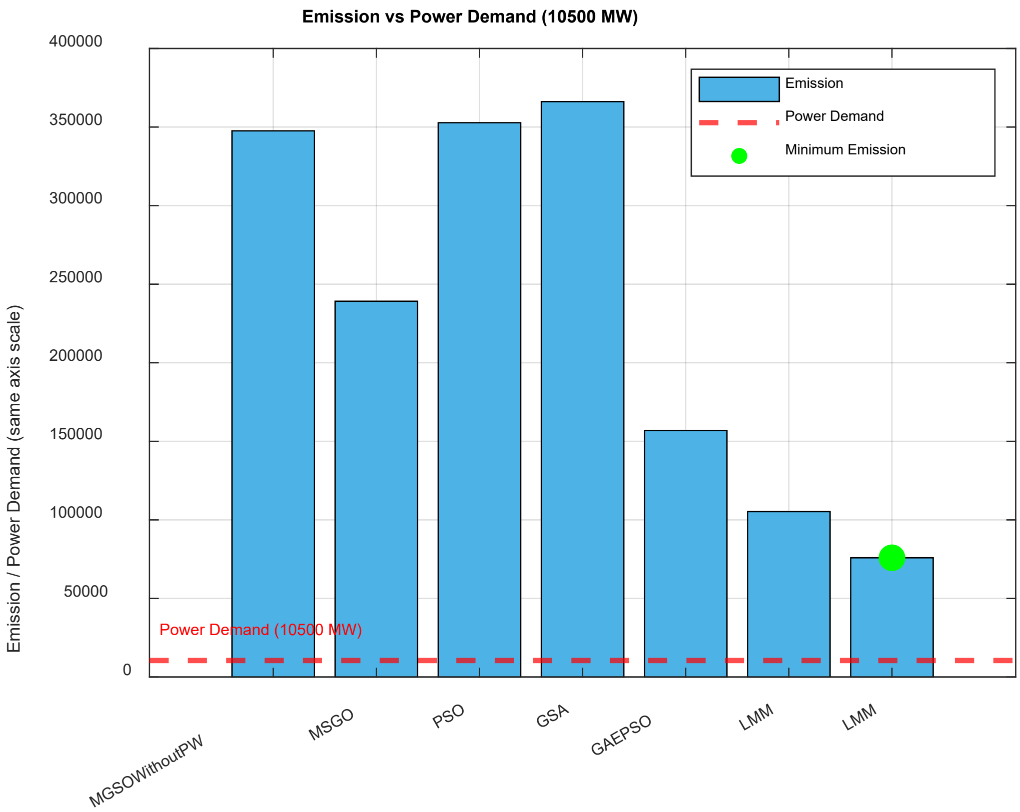



**Figure 3.9:** **Values of emission and power demand for various algorithms using a 40-unit system in relation to the developed algorithm.**

## 3.4 Conclusions

This chapter employs the Lagrange multiplier method to provide a successful solution to the wind-thermal economic emission dispatch problem. Fuel and emission costs are minimised in the bi-objective problem for economic emission dispatch optimisation. It also addresses transmission line losses, emissions, and inequality constraints. Three test systems, 6-unit, 10-unit, and 40-unit, are implemented to evaluate the effectiveness of the algorithm. Compared to other algorithms, the fuel cost and emissions for 6-unit systems are 3-6% and 1-3% lower, respectively. Additionally, compared with other optimisation techniques, the 10-unit system's fuel cost and emissions are 1-2% and 1-2.89%, respectively. Lastly, fuel consumption and emissions for the bigger system have been decreased by 1-6% and 1-32%, respectively.

Across all test systems, the Lagrange optimisation approach effectively converges to an optimal solution by selecting the best initial values for the lambda parameters. The resulting computational times for the 6-unit, 10-unit, and 40-unit models were 11.042308s for 84 iterations, 11.657372s for 1013 iterations, and 16.145170s for 72 iterations. In comparison, the optimal values for the 6-unit, 10-unit, and 40-unit systems are found to fall within the expected values reported in the literature for fuel cost, emissions, and total generation in the WTEED problem. It can also be seen that, irrespective of load demand or the types of generators used, if the initial guess of lambda is chosen carefully, the Lagrange multiplier always finds an optimal solution quickly and precisely. Furthermore, the results of the LM algorithm have been contrasted with those of other heuristic algorithms documented in the recent literature. Compared with heuristic approaches documented in the literature, the conventional LM algorithm may readily achieve optimal results in reducing fuel consumption and pollution. Considering straightforward, convex scenarios, Lagrange's classical methods are effective and precise. Chapter 4: Solving the Wind-Thermal Economic Emission Dispatch (WTEED) problem using Particle Swarm Optimisation (PSO) and comparing the results of heuristic and classical optimisation techniques.

# CHAPTER FOUR

## WIND-THERMAL ECONOMIC EMISSION DISPATCH(WTEED) PROBLEM USING PARTICLE SWARM OPTIMIZATION(PSO)

### 4.1 Introduction

This chapter solves the problem of WTEED using the PSO. Many researchers have solved the WTEED using PSO. In the work of (Mishra & Shaik, 2024b; Secui et al., 2024b; Zhang et al., 2019)The problem for WTEED is solved by minimizing both the fuel cost and emission using the price penalty factor h. Several price factors are used in solving the economic emission dispatch problem(Krishnamurthy & Tzoneva, 2012). In this chapter, a min-min penalty factor is adopted. The impact of varying wind power on the solution of the WTEED problem is compared across different IEEE 30-bus systems with 6-unit, 10-unit, and 40-unit generators. All three case studies are compared with the existing literature solution values using the developed PSO algorithm. The PSO algorithm is also compared with the classical Lagrange multiplier method developed in Chapter 3 for solving the WTEED problem.

A description of the PSO algorithm is presented in 4.2. In 4.3, the PSO method for solving the CEED problem is developed. The application of the PSO method and its comparison with the Lagrange method for IEEE power systems models are illustrated in 4.4. Sections 4.5 and 4.6 present the discussion and conclusions.

### 4.2 PSO algorithm

Kennedy and Eberhart developed the particle swarm optimisation technique in 1995. It is an optimisation based on how the swarm's species behaves. It provides a population-based search strategy in which species, referred to as particles, navigate an open, multidimensional space and change their positions over time.

Based on both its own and the other particle's experiences, the particle positioned itself in the solution space utilising the best position that both of them had encountered. Based on past experiences, the swarm follows the nearby particle. Every particle in the swarm travels around the search space at a specific speed that varies based on its prior experiences.

Let a particle coordinate (position) and its corresponding flight speed (velocity) in a search space denotd by p and v, respectively.The $i^{th}$particle is preserved as a volume les particle assumed by $p_i = (p_{i1}, p_{i2} \dots p_{id})$ and the best position as $p^{best}{}_i = (p^{best}{}_{i1}, p^{best}{}_{i2} \dots\ p^{best}{}_{id})$.The index of the best particle amongst others is given a name called global best $g_d{}^{best}$. The change of velocity of the $i^{th}$ particle is given by $v_i = (v_{i1}, v_{i2} \dots v_{id})$. Equations (4.1) and (4.2) provide the immediate velocity and the distance between $p_{id}{}^{best}$ and $g_d{}^{best}$, respectively, which may be utilised to determine the updated velocity and location of each particle.

$$v^{t+1}{}_{id} = \omega v^t{}_i + c_1 rand()(p^{best}{}_{id} - p^t{}_{id}) + c_2 rand()(p^{best}{}_{id} - p^t{}_{id}) \quad (4.1)$$

$$p_{id}{}^{t+1}{}_i = p^t{}_{id} + v^{t+1}{}_{id} \quad (4.2)$$

Where

$c_1$ Self-assurance ranges between 1.5 and 2, and

$c_2$ Swarm range between 2 and 2.5.

$(p^{best}{}_{id} - p^t{}_{id})$ and $(p^{best}{}_{id} - p^t{}_{id})$ are a particle memory influence and a swarm influence.

The proper balance between global and local search can be achieved by selecting the inertia weight factor $\omega$, to be between 0.9 and 0.4 during the search, and can be given by the following expression: $\omega = \omega_{\max} - \left[\frac{\omega_{\max} - \omega_{\min}}{iter_{\max}}\right] iter$

Where,

| | |
|---|---|
| $\omega$ | The inertia weight factor |
| $\omega_{\max}$ and $\omega_{\min}$ | Maximum and minimum weighting factors |
| $Iter$ and $Iter_{max}$ | Current and maximum iterations. |

## 4.3 WTEED problem solution using the developed PSO algorithm

In Chapter 3, the WTEED problem is studied, formulated, and solved using LMM. The problem is formulated using the total fuel cost equation for thermal and wind power systems as given in equation (2.1). The fuel cost for conventional and wind power generators is given by equations (2.2) and (2.3). This is subject to equality and inequality constraints given in equations (2.4) and (2.6). The wind power system is further analyzed using the probability distribution function (PDF), yielding equations (2.46) and (2.51). The emission function is given by equation (2.52), and the price penalty factors (3.1)-(3.3) are used to formulate the single-objective function for the WTEED problem. The WTEED problem is solved using a heuristic optimisation algorithm, PSO, tailored to incorporate the specifics of the WTEED problem. It is needed to associate the structure of the WTEED problem with that of the velocity and position Equations (4.1) and (4.2) adopted by the PSO algorithm. This can be achieved with the following steps:

- The number of members in a distinct particle inside the swarm is equivalent to the number of generators. The active power generated by conventional and wind power producers is represented by the positions of particle members in the wind thermal economic dispatch problem.
- The velocities are parameters that represent the active power, even if they are utilized for finding within the boundaries.
- The swarm's particle population is determined to be $N_P$. Equations (2.1) to (2.6) and (2.46) to (2.52) provide the developed PSO algorithm for solving the combined economic dispatch problem:

**Step1:** Set the PSO initial values for the acceleration constants and inertia weight. The maximum number of iterations ($Iter^{max}$) and uniform random values (rand1, rand2) for c1 and c2.

**Step 2:** Employing the generator limit constraints in Equation (3.20), determine the minimum and maximum initial velocities, which are provided in Equation (4.3) as follows:

$$-0.5P_{pi}^{min} \leq V_{pi} \leq +0.5P_{pi}^{max}, p = \overline{1, N_p}, i = \overline{1, n-1} \tag{4.3}$$

Where

$N_P$ Quantity of particles in the swarm

$n$ The total number of generators equals the number of members in one particle.

The particle position and velocity are calculated for $(n-1)$ generators, since the first generator is acknowledged as a slack one.

**Step 3:** Apply equation (4.4) to compute the initial velocity of every particle, with the exception of the slack bus generator.

$$V_{pi} = V_{pi}^{min} + rand()(V_{pi}^{max} - V_{pi}^{min}), p = \overline{1, N_p}, i = \overline{1, n-1} \tag{4.4}$$

Where

$V_{pi}^{min}, V_{pi}^{max}$ Minimum and maximum velocities previously determined

**Step 4:** Employ Equation (4.5) to determine the particle members' starting positions.

$$P_{pi} = P_{pi}^{min} + rand()(P_{pi}^{max} - P_{pi}^{min}), p = \overline{1, N_p}, i = \overline{1, n-1} \tag{4.5}$$

Wind power is added to equation (4.5) using equations (2.46) to (2.52) to obtain the active power for the WTEED problem. Then this is verified if they are within the given limits in equation(4.20), and is represented by the following equation(4.6)

$$\begin{Bmatrix} P_{pi}^{min}, & P_{pi} \leq & P_{pi}^{min} \\ P_{pi}^{max}, & P_{pi} \geq & P_{pi}^{max} \\ P_{pi}, & P_{pi}^{min} \leq P_{pi} \leq & P_{pi}^{max} \end{Bmatrix}, p = \overline{1, N_p}, i = \overline{1, n-1} \tag{4.6}$$

The power system's buses are divided into three categories: load (PQ), generator (PV), and slack buses. Any bus connected to a generator with the greatest generating capacity is referred to as a slack bus in a power system. The system's voltage and angle are referenced by the slack bus. The slack bus's real and reactive powers are uncontrollable; the actual or reactive power necessary to balance the system's power flows is provided by the bus.

Generation scheduling is used to solve the economic dispatch problem without accounting for the actual power output of the slack bus generator. In the PSO algorithm, the slack bus serves to meet the power balance requirement specified in Equation (2.4). Step 5 outlines the process for determining the real power of the slack bus generator in the PSO algorithm.

**Step 5:** The slack bus generator is considered to have the most power-generating capability among dependent generators. Equation (4.7) provides the power balance constraint that may be used to calculate the initial active power $P_{pd}$ using the bus voltage magnitude and phase angle as a reference(Krishnamurthy & Tzoneva, 2012):

$$P_{pd} + \sum_{\substack{i=1 \\ i \neq d}}^{n} P_{pi} = \left\{ \sum_{\substack{i=1 \\ i \neq d}}^{n} \sum_{\substack{j=1 \\ j \neq d}}^{n} P_{pi} B_{ij} P_{pj} + \sum_{\substack{j=1 \\ j \neq d}}^{n} P_{pj} \left( B_{jd} + \right. \right.$$

$$\left. \left. B_{dj} \right) P_{pd} + B_{dd} P_{pd}^2 + \sum_{\substack{i=1 \\ i \neq d}}^{n} B_{io} P_{pi} + B_{do} P_{pd} + B_{oo} + P_D \right\}, p = \overline{1, N_p,} \tag{4.7a}$$

Where

$P_{pd}$ Is the power generated by the slack bus?

$P_D$ Is the entire power demand of the power system

$\sum_{\substack{i=1 \\ i \neq d}}^{n} P_{pi}$ Is the power system's total active power, except for the slack bus power.

, and is

given by. $\sum_{\substack{i=1 \\ i \neq d}}^{n} P_{pi} = \left[ P_{T1}, P_{T2} \dots P_{TN_T}, P_{W1}, P_{W2} \dots P_{WN_w} \right],$ (4.7b)

Where,

$P_{TN_T}$ Is the active power produced by thermal power generators.

$P_{WN_w}$ Is the active power produced by wind power generators.

By using the quadratic form, Equation (4.7) can be rewritten, where the $P_{pd}$ is the unknown variable in the equation.

$$XP_{pd}^2 + YP_{pd} + Z = 0 \tag{4.8}$$

Where

$$X = B_{dd} \tag{4.9}$$

$$\sum_{\substack{j=1 \\ j\neq d}}^{n} P_{pj}\left(B_{jd} + B_{dj}\right) + B_{do} - 1 \tag{4.10}$$

$$Z = \sum_{\substack{i=1 \\ i\neq d}}^{n} \sum_{\substack{j=1 \\ j\neq d}}^{n} P_{pi}B_{ij}P_{pj} + \sum_{\substack{i=1 \\ i\neq d}}^{n} B_{io}P_{pi} + B_{oo} + P_D - \sum_{\substack{i=1 \\ i\neq d}}^{n} P_{pi} \tag{4.11}$$

Equation (4.8) positive root can be determined as follows:

$$P_{pd} = \frac{-Y+\sqrt{Y^2-4XZ}}{2X}, where\ Y^2 - 4XZ \geq 0 \tag{4.12}$$

The actual power vector is now constructed as $P_p = \left[P_{pd}, P_{pi}, i = 1, n, i \neq d\right]$ $where\ p = \overline{1, N_p}$

**Step 6:** Determine the following objective functions for the particles' starting positions.

Fuel cost function

$$F_{Cp} = \sum_{i=1}^{n}\left(a_i P_{pi}^2 + b_i P_{pi} + c_i\right), p = \overline{1, N_p} \tag{4.13}$$

Emission function

$$F_{Tp} = \sum_{i=1}^{n}\left(d_i P_{pi}^2 + e_i P_{pi} + f_i\right), p = \overline{1, N_p} \tag{4.14}$$

Combined economic emission function

$$F_{CT} = \sum_{i=1}^{n}\left(a_i P_{pi}^2 + b_i P_{pi} + c_i\right) + h_{pi}\left(d_i P_{pi}^2 + e_i P_{pi} + f_i\right), p = \overline{1, N_p} \tag{4.15}$$

Where the calculation of the Min-Max price penalty factor is

$$h_{pi} = \frac{a_i P_{pi,max}^2 + b_i P_{pimax} + c_i}{d_i P_{pi,max}^2 + e_i P_{pi,max} + f_i}, p = \overline{1, N_p}, i = \overline{1, n} \tag{4.16}$$

The fuel cost values $F_{TP}$ of all the particles are organised in ascending order. The value at the highest point of the ascending sequence is chosen as the $F_T^{best}$.

**Step 7:** Choose the global best starting position and the best initial position as follows:

The initial particle locations in the swarm are assumed to be optimal.

$P_p^{best} = min P_p^{best}, p = \overline{1, N_p}, i = \overline{1, n}$ .

- The best position out of all the best particles $minP_p^{best},\ p = \overline{1, N_p}$ is taken as

$G_p^{best} = minP_p^{best}, p = \overline{1, N_p}$

$l^{th}$ step of the iteration process commences, where $l = l + 1$

**Step 8**:Applying equation (4.17) to calculate the new velocity.

$$V_{pi}^{new^l} = \omega.V_{pi}^{l-1} + c1 * rand1()\left(P_p^{best^{l-1}} - P_{pi}^{l-1}\right) + c1 * rand1()\left(G^{best^{l-1}} - P_{pi}^{l-1}\right),$$
$$p = \overline{1, N_p}, i = \overline{1, n-1} \quad (4.17)$$

Check the limits on the minimum and maximum velocity values.

$$\text{If } V_{pi}^{new^l} > V_{pi}^{max^{l-1}}, V_{pi}^{new^l} = V_{pi}^{max^{l-1}} \text{ and}$$
$$\text{If } V_{pi}^{new^l} < V_{pi}^{min^{l-1}}, V_{pi}^{new^l} = V_{pi}^{min^{l-1}}, p = \overline{1, N_p}, i = \overline{1, n-1} \quad (4.18)$$

**Step 9:** Applying Equation (4.19), determine the generator's new location within the particles.

$$\text{If } P_{pi}^{new^l} = P_{pi}^{l-1} + V_{pi}^{new^l}, p = \overline{1, N_p}, i = \overline{1, n} \quad (4.19)$$

**Step 10:** Employ the constraint Equation (4.6) to validate the generators' new positions within the particles as follows

$$P_{pi}^{new^l} = \begin{Bmatrix} P_{pi}^{min}, & P_{pi}^{new^l} \le & P_{pi}^{min} \\ P_{pi}^{max}, & P_{pi}^{new^l} \ge & P_{pi}^{max} \\ P_{pi}^{new}, & P_{pi}^{min} \le P_{pi}^{new^l} \le & P_{pi}^{max} \end{Bmatrix}, p = \overline{1, N_p}, i = \overline{1, n-1} \quad (4.20)$$

**Step 11:** Determine the new real power of the slack bus generator. Utilising the formula from step 5, establish the new real power vector $P_{pd}^{new^l}$. Employing the constraint Equation (4.6), determine the slack bus generators' new position within the particles.

$$P_{pd}^{new^l} = \begin{Bmatrix} P_{pd}^{min}, & P_{pd}^{new^l} \le & P_{pd}^{min} \\ P_{pd}^{max}, & P_{pd}^{new^l} \ge & P_{pd}^{max} \\ P_{pd}^{new}, & P_{pd}^{min} \le P_{pd}^{new^l} \le & P_{pd}^{max} \end{Bmatrix} \quad (4.21)$$

**Step 12:** For the $l^{th}$ iteration, the entire active power vector is:

$$P_{pd}^{new^l} = \left(P_{pd}^{new^l}, P_{pi}^{new^l}, i \ne d, i = \overline{1, n}\right), p = \overline{1, N_p} \quad (4.22)$$

**Step 13:** Compute the new objective functions $F_T^{new}$ with Step 6

**Step 14:** Validate the new objective function $F_T^{new}$ as explained below

$$\boldsymbol{If}\ F_T^{new^l} < F_T^{best^{l-1}} then\ F_T^{best^l} = F_T^{new^l} and\ P_{pi}^{best^l} = P_{pi}^{new^l}$$

$$\text{Else } If\ F_T^{best^l} = F_T^{best^{l-1}} and\ P_{pi}^{best^l} = P_{pi}^{best^{l-1}} \quad (4.23)$$

$$G^{best^l} = P_p^{best^l}, p = \overline{1, N_p}$$

where $l$ is the number of iterations

The best solution $G^{best}$ is only one for the whole system. The best solution per particle is only one $P_p^{best} = minP_{pi}, i = \overline{1,n}$.

The best solution for the whole system is $P_p^{best} = \left[P_1^{best}, P_1^{best}, \dots., P_{N_p}^{best}\right]$. Then

$G^{best} = minP_p^{best^l}, p = \overline{1, N_p}$

**Step 15:** Repeat steps 5-13 until the maximum number of iterations is reached. The PSO flowchart is shown in Figure 4.1 below.

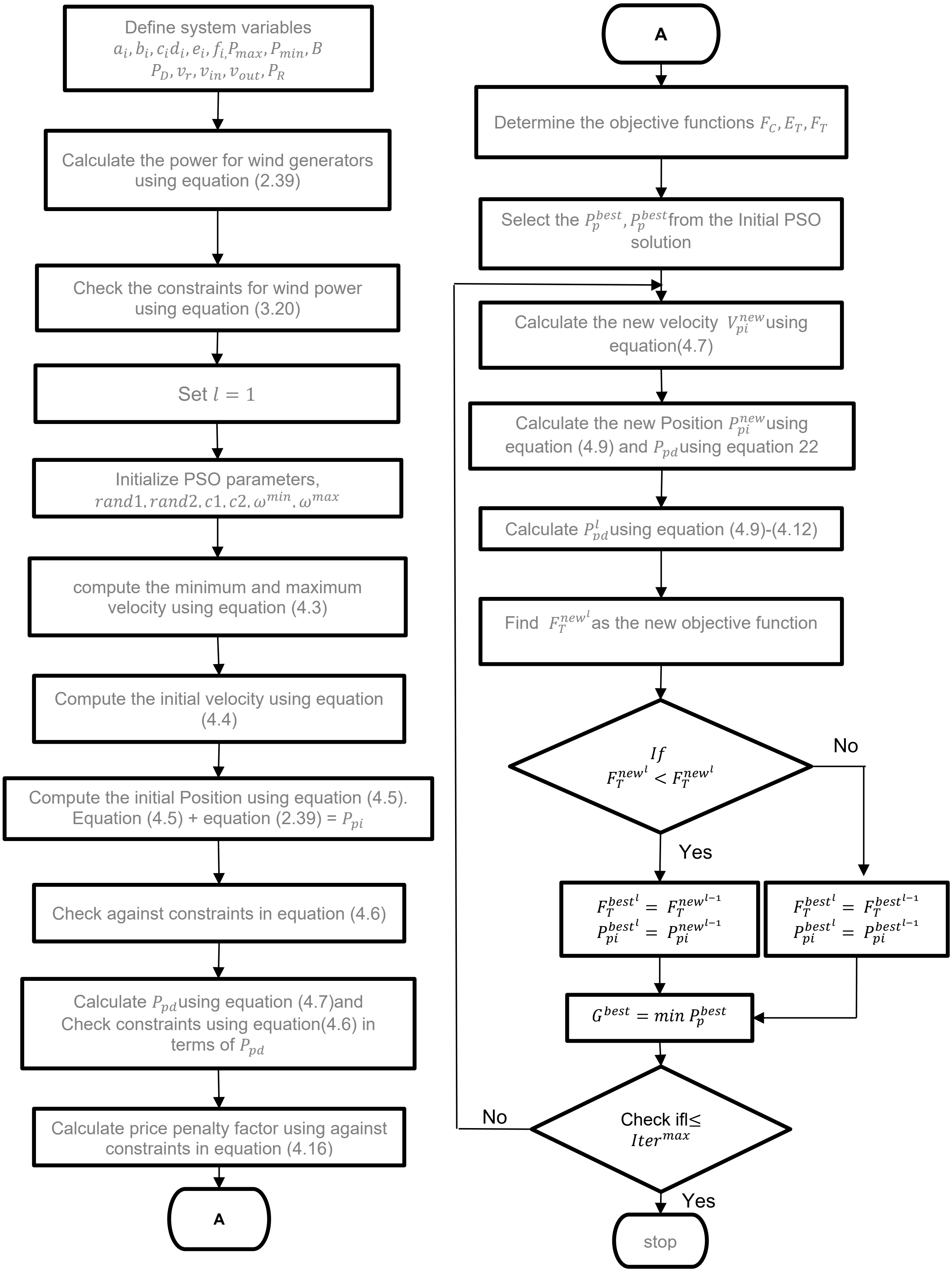



**Figure 4.1:** PSO algorithm for the WTEED problem solution flowchart.

## 4.4 Application of the PSO algorithm to solve the WTEED problem

The PSO algorithm is used to solve the WTEED problem. The problem is tested using three different case studies: the IEEE 30-bus system with 6 units, the IEEE 10-unit system, and the IEEE 40-unit system. These are discussed separately in 4.4.1, 4.4.2, and 4.4.3, respectively.

### 4.4.1 Test system 1: IEEE 30 bus system with six generation units

The problem of solving WTEED using the PSO algorithm is addressed using a system comprising six thermal generators and one wind turbine cluster. The fuel cost parameter are formed with the ones reported in (Krishnamurthy & Tzoneva, 2012) , and (Gnanadass, 2005), and for wind power, the coefficients are those reported in (Damodaran & Sunil Kumar, 2018; Liu & Xu, 2010). The bi-objective optimisation problem is converted to a single problem by using the max-max price penalty factor as given in equations (3.1)-(3.3)(Krishnamurthy & Tzoneva, 2012). Several power demand values are presented in Table 4.1 in order to test the system's behaviour to changes in power demand. The software is provided in Appendix B1, and the MATLAB script is named WTEED_casePSO6units.m. The solution of the WTEED problem is shown in Table 4.1. The real power of the thermal(P1-P6) units, wind (Pw) generators, transmission losses ( $P_L$ ) in [MW], fuel cost ($F_C$) for thermal generators, total emission ($E_T$) in [ kg/h], and CEED in [Kg/h].

**Table 4.1:** **The wind-thermal economic emission dispatch problem using PSO method with various power demands**

| PD[MW] | 125 | 150 | 175 | 225 | 250 | 283.4 |
|---|---|---|---|---|---|---|
| P1[MW] | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 |
| P2[MW] | 5.00 | 5.00 | 5.00 | 10.34 | 16.11 | 41.87 |
| P3[MW] | 5.00 | 5.00 | 5.00 | 5.00 | 5.00 | 10.13 |
| P4[MW] | 5.00 | 26.22 | 5.00 | 91.44 | 120.00 | 5.00 |
| P5[MW] | 5.00 | 5.00 | 5.00 | 5.13 | 5.00 | 9.92 |
| P6[MW] | 5.00 | 5.00 | 52.39 | 12.45 | 5.00 | 13.09 |
| PwMW] | 100.2 | 100.2 | 100.2 | 100.20 | 100.2 | 200.00 |
| PL[MW] | 1.00 | 1.400 | 2.6 | 4.50 | 6.30 | 4.00 |
| $F_C$[$/h] | 50,820 | 53,0081 | 58,1999 | 62,2608 | 65,4475 | 59,5914 |
| $E_T$[Kg/h] | 28.40 | 28.00 | 27.40 | 29.10 | 31.30 | 25.8 |

The values for transmission losses ($P_L$) in MW, fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h] for the WTEED problem shown in Table 4.1 above.

The wind-thermal economic emission dispatch problem using the PSO algorithm is compared with other optimization algorithm (Zhang et al., 2019). The Lagrange multiplier method developed in Chapter 3, shown in Table 4.2 below, is compared with various optimization algorithms.

**Table 4.2:** **PSO and LMM compared to other optimisation algorithms for the WTEED problem, PD = 283.4 MW**

| Reference | (Jiang et al., 2019) | | | Developed algorithms | |
|---|---|---|---|---|---|
| **Algorithm** | **GSA** | **GPSOA** | **GA** | **LMM** | **PSO** |
| P1[MW] | 49.21 | 31.65 | 36.62 | 19.76 | 5.00 |
| P2[MW] | 46.21 | 45.22 | 42.58 | 41.31 | 41.87 |
| P3[MW] | 36.69 | 42.49 | 30.31 | 51.05 | 10.13 |
| P4[MW] | 33.32 | 38.37 | 31.36 | 89.58 | 5.00 |
| P5 [MW] | 43.77 | 43.42 | 53.44 | 49.58 | 9.92 |
| P6[MW] | 53.51 | 47.62 | 43.52 | 35.09 | 13.09 |
| Pw [MW] | 20.69 | 34.63 | 45.57 | 200.200 | 200.20 |
| $P_L$[MW] | - | - | - | 3.9600 | 4.00 |
| $F_C$[$/h] | 66123.00 | 66134.00 | 66582.00 | 50589.00 | 59591.4 |
| %Deviation of Fc for PSO in relation to the literature | 2.59 | 2.60 | 2.77 | 4.08 | 3.01 Average |
| $E_T$[Kg/h] | 23.17 | 24.26 | 26.18 | 19.6400 | 25.8 |
| %Deviation of $E_T$ for PSO in relation to the literature | 2.69 | 1.54 | 0.37 | 6.79 | 2.8475 Average |

The WTEED problem is solved considering a load demand of 283.4MW and a find warm consisting of 200 identical wind turbines, each with 1.5 MW power capacity, for a total wind farm installed capacity of 200 MW.(Liu et al., 2018). The PSO algorithm is compared with Genetic Algorithm (GA), Gravitational Search Algorithm (GSA), and a hybrid Gravitational Particle Swarm Optimization Algorithm (GPSOA).

#### 4.4.1.1 The IEEE30 bus system with 6-unit results discussion

This system has utilized 50 available wind turbines that supply the power dispatch problem with 200MW. The developed algorithm for a 6-unit system is used and compared with LMM and other optimisation algorithms as presented in Table 4.2. The fuel cost obtained for the GSA, GPSOA, GA, and LMM algorithms is 2.59%, 2.60%, 2.77%, and 4.08%, respectively, compared to the PSO algorithm. The proposed algorithm is 1–3% better than other algorithms used to solve the wind–thermal economic emission dispatch problem. The resultant NOx emissions are 2.69%, 1.54%, 0.37%, and 6.79%, 0.26%, compared to the proposed PSO algorithm. The PSO method has effectively reduced pollutant levels by 1–2.8% compared to other optimization algorithms. The table shows that fuel cost and emissions are lower with Lagrange's method than with the other methods. This can also be shown graphically, as presented in Figures 4.2 and 4.3 below. The following graphs show the relationships between power demand, fuel cost, and emissions for the various methods presented in Table 4.2 above.

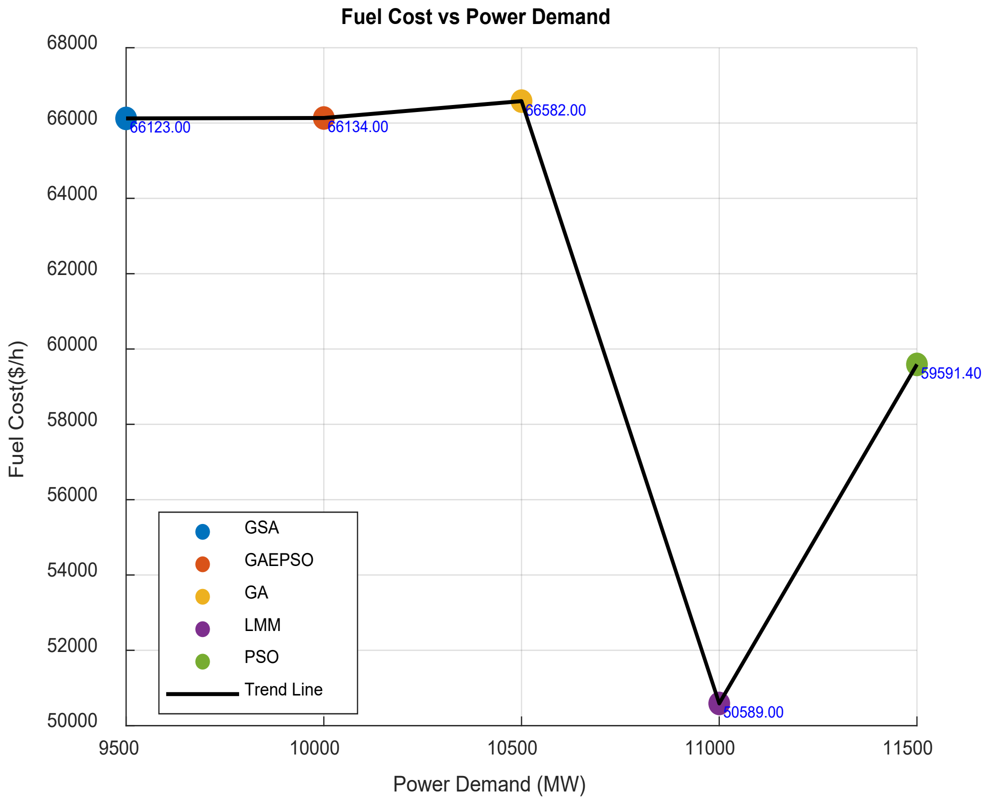



**Figure 4.2:** Values of fuel cost and power demand for various algorithms using a 6-unit system compared to the developed PSO

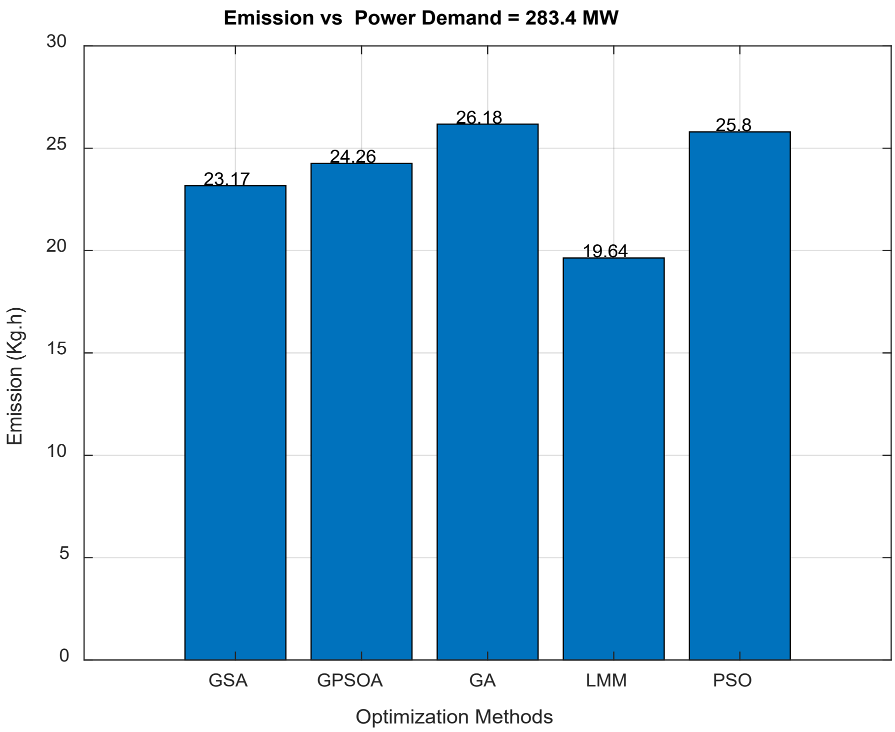



**Figure 4.3:** Values of emission and power demand for various algorithms using a 6-unit system compared to the developed PSO

### 4.4.2 Test system 2: IEEE 30 bus system with 10 generation units

This study of the WTEED power system problem using PSO includes 10 thermal generators and one wind turbine cluster. The cluster consists of wind turbines situated in a geographic area, characterised by a group of equal turbines in a large wind farm. The fuel cost, emission, and loss coefficients are constructed based on the ones reported on (Basu, 2011b)Moreover, the results are presented in Tables 3.6 and 3.7. The test system uses the same wind parameters as reported in Test System I. The multi-objective optimisation problem is converted to a single problem by using the min-max price penalty factor as in test system I. The study focuses on the 2000MW power demand as presented in(Mishra & Shaik, 2024b). Several power demand values ranging from 650 to 2000 [MW] are used to test the system's behaviour to changes in power demand. The software is provided in Appendix B2, and the MATLAB script is named WTEED_casePSO10units. m. The solution of the WTEED problem is shown in Table 4.3. The table illustrates the real power of the thermal(P1-P10) units, wind (Pw) generators, and transmission losses ($P_L$) in [MW], fuel cost ($F_C$) for thermal generators, total emission ($E_T$) in [ kg/h], and CEED in [Kg/h].

**Table 4.3: The wind-thermal economic emission dispatch problem using PSO with various power demands for 10-unit generators**

| PD[MW] | **650** | **800** | **1000** | **1200** | **1500** | **2000** |
|---|---|---|---|---|---|---|
| P1[MW] | 10 | 10.00 | 10.00 | 26.43 | 14.1450 | 37.6848 |
| P2[MW] | 20 | 20.00 | 67.02 | 42.78 | 37.3342 | 70.8940 |
| P3[MW] | 47 | 47.00 | 117.63 | 58.55 | 104.4475 | 81.0499 |
| P4[MW] | 20 | 20.49 | 21.58 | 96.21 | 72.6856 | 128.5784 |
| P5[MW] | 50 | 122.01 | 90.53 | 92.29 | 87.2873 | 108.1157 |
| P6[MW] | 70 | 70.00 | 70.00 | 102.29 | 150.3286 | 146.7899 |
| P7[MW] | 60 | 60.00 | 60.00 | 108.19 | 178.7130 | 252.6337 |
| P8[MW] | 70 | 82.73 | 200.01 | 70.55 | 247.4480 | 265.0825 |
| P9[MW] | 135 | 135.00 | 135.00 | 229.37 | 323.7965 | 470.0000 |
| P10[MW] | 150 | 150.00 | 150.00 | 314.43 | 235.4756 | 441.0383 |
| Pw[MW] | 100.166 | 100.166 | 100.166 | 100.17 | 100.166 | 100.166 |
| PL[MW] | 15.14 | **17.40** | 21.93 | 41.26 | 51.827 | 102.033 |
| $F_C$[$/h] | 37137.345 | 42392.449 | 52214.646 | 62202.01 | 78621.468 | 107838.272 |
| $E_T$[Kg/h] | 1427.308 | 1508.206 | 1898.959 | 1971.98 | 2553.717 | 3988.590 |
| CEED[Kg/h] | - | 44008.891 | 54113.606 | 64173.99 | 81175.186 | 111826.862 |

The optimum values of transmission losses ($P_L$) in MW, fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h] for the WTEED problem shown in Table 4.3 above. The WTEED problem outputs obtained using the PSO algorithm are compared with those of multiple optimization algorithms reported in (Mishra & Shaik, 2024b) and are shown in Table 4.4. The solution to the problem for WTEED is considered with a load demand of 2000MW and a wind farm consisting of 50 identical wind turbines. The developed PSO algorithm is compared with the developed Lagrange multiplier method from Chapter 3, as well as with Moth-Flame Optimisation (MFO), the Whale

Optimisation Algorithm (WOA), the Search and Rescue Algorithm (SAR), and the Genetic Algorithm (GA) (Mishra & Shaik, 2024b).

**Table 4.4:** **PSO compared to the LM method and other optimization algorithms for the WTEED Problem applied to a 10-unit system, PD = 2000[MW]**

| Algorithm | MFO | WOA | SAR | GA | Developed LMM | Developed PSO |
|---|---|---|---|---|---|---|
| P1[MW] | 37.16 | 46.96 | 53.49 | 34.17 | 107.0710 | 37.6848 |
| P2[MW] | 25.84 | 68.61 | 77.10 | 63.49 | 104.4629 | 70.8940 |
| P3[MW] | 120 | 69.24 | 64.41 | 95.75 | 108.4683 | 81.0499 |
| P4[MW] | 93.90 | 129.21 | 83.05 | 121.67 | 107.9097 | 128.5784 |
| P5[MW] | 160 | 50 | 158.73 | 145.83 | 105.2629 | 108.1157 |
| P6[MW] | 240 | 226 | 239.50 | 229.86 | 126.3436 | 146.7899 |
| P7[MW] | 244.61 | 298.21 | 275.88 | 281.21 | 244.2553 | 252.6337 |
| P8[MW] | 286.25 | 337.93 | 332.28 | 307.35 | 276.0585 | 265.0825 |
| P9[MW] | 470 | 390.40 | 409.12 | 391.63 | 407.7432 | 470.0000 |
| P10[MW] | 404.92 | 468 | 388.56 | 410.56 | 410.7996 | 441.0383 |
| PW[MW] | 30 | 30 | 30 | 30 | 100.166 | 100.166 |
| Fc[$] | 116690.89 | 114678.84 | 114106.19 | 115802.24 | **107533.05** | **107838.272** |
| %Deviation of Fc for PSO in relation to the literature | 4.10 | 3.17 | 2.91 | 3.69 | 0.14 | 2.79 Average |
| PL[MW] | 83.70 | 84.44 | 82.81 | 83.9 | **98.490** | **102.033** |
| ET[T] | 4080 | 4220 | 4103.3 | 4390 | **3967.871841** | **3988.590** |
| %Deviation of $E_T$ for PSO in relation to the literature | 1.15 | 2.90 | 1.44 | 5.03 | 0.26 | 2.16 Average |
| CEED[$] | 239158.66 | 244987.20 | 226506.78 | 231903.60 | 111500.921 | 111826.862 |

#### 4.4.2.1 The IEEE 30 bus system with 10 units results discussion

The PSO algorithm is applied to solve the economic emission dispatch problem with wind power, using 10 conventional generators and 50 available wind turbines across 50 identical wind farms. The fuel cost obtained for the MFO, WOA, SAR, GA, and LMM algorithms are 4.10%, 3.17%, 2.91%, 3.69%, and 0.14%, respectively, compared to the PSO algorithm. The proposed algorithm is 1–3% superior to other algorithms for solving the wind–thermal economic emission dispatch problem. The resultant NOx emissions are 1.15%, 2.90%, 1.44%, 5.03%, and 0.26% compared to the proposed LM algorithm. The LM method has effectively reduced pollutant levels by 1–2% compared to other optimization algorithms. The table shows that fuel cost, emissions, and CEED values are lower with Lagrange's method than with the other methods. Nevertheless, the obtained solutions are not the same as Lagrange's, since PSO generates a random solution for each execution. Fuel cost, emissions, and CEED are shown graphically in Figures 4.4, 4.5, and 4.6 below.

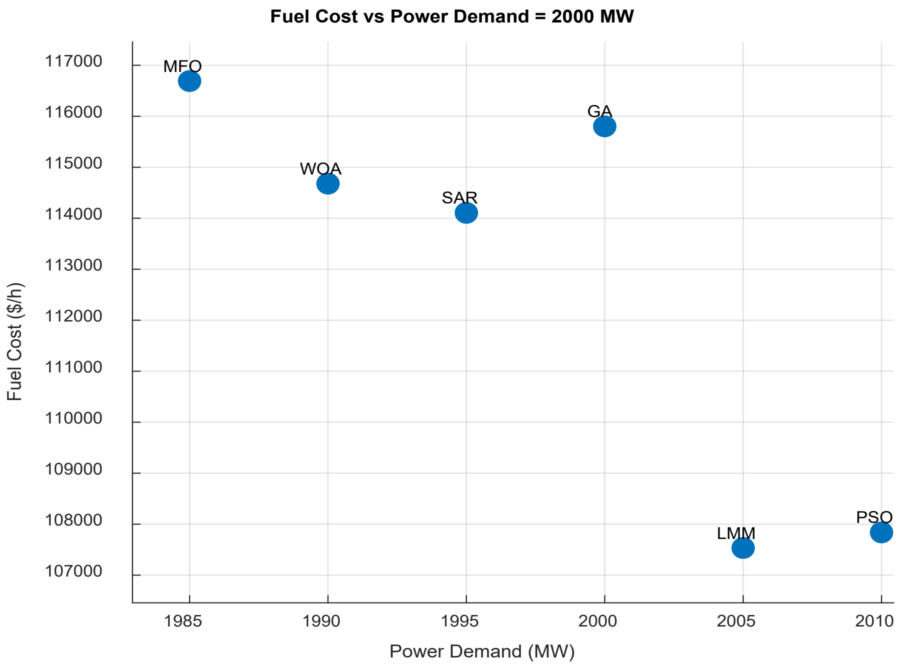



**Figure 4.4:** Values of fuel cost and power demand for various algorithms applied to a 10-unit system, and compared to the developed PSO algorithm

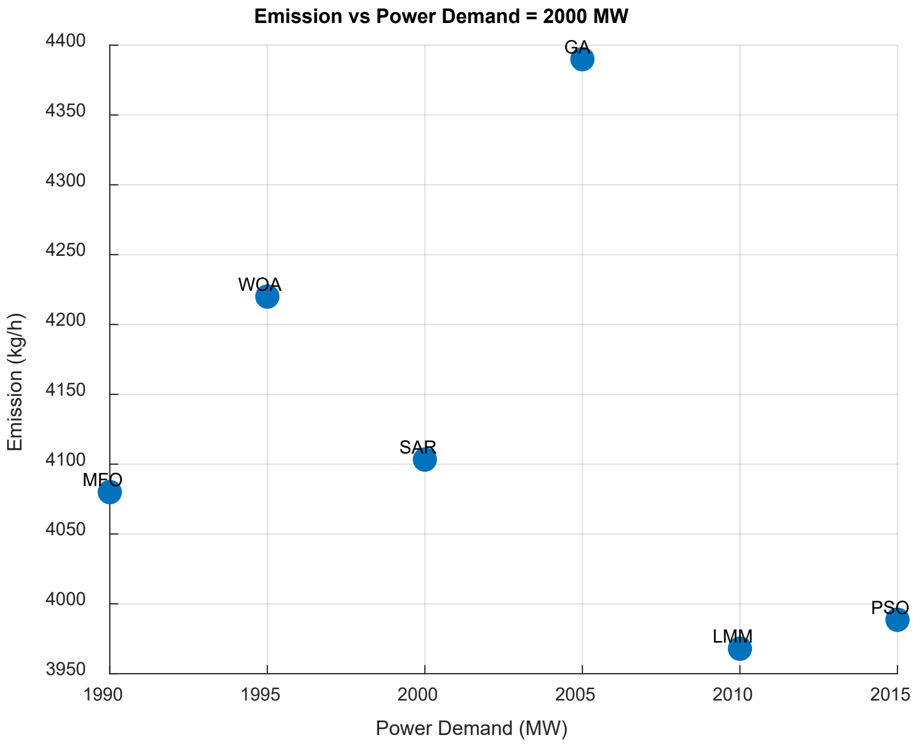



**Figure 4.5:** Values of emission and power demand for various algorithms applied to a 10-unit system, and compared to the developed PSO algorithm

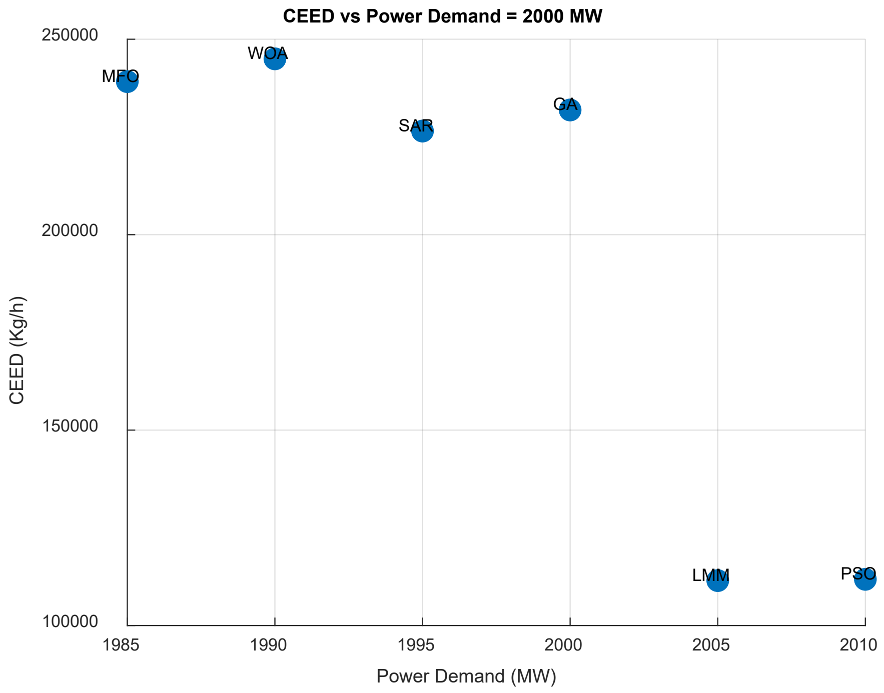



**Figure 4.6:** **Values of CEED and power demand for various algorithms applied to a 10-unit system in relation to the developed PSO algorithm**

### 4.4.3 Test System 3: IEEE 30 bus system with 40 generation units

The solution for the WTEED is verified using a power system with 40 thermal power generators, a power demand of 10500MW, and transmission losses, excluding VPLE. Power constraints, fuel cost coefficients, pollution coefficients, and B-loss coefficients are among the power system constraints that originate from(Secui et al., 2024b)The coefficients are illustrated in Tables 3.9 and 3.10, respectively.

As noted in test system two, the cluster consists of land-based wind turbines, including a collection of identical turbines in a sizable wind farm. Similar to test systems I and II above, the max-max price penalty factor is used to convert the bi-objective optimisation problems into a single optimisation power system problem.

The PSO algorithm is contrasted with various heuristic algorithms, i.e., (GSA, GAEPSO)(Jiang et al., 2015) and MSGO(Secui et al., 2024b) as well as with the developed Lagrange Multiplier Method. The evaluation is completed for the best cost and emission. The system response to variations in power demand can be tested using a range of power demand values. Table 4.5 illustrates the solution to the WTEED problem. The actual power values for thermal (P1–P40) units, wind (Pw) generators, fuel cost ($F_C$) for thermal generators, transmission losses ($P_L$) in [MW], total emissions ($E_T$) in [kg/h], and CEED in [Kg/h]. The software is provided in Appendix B3, and the MATLAB script is termed WTEED_casePSO40units.m

**Table 4.5:** **Various power demands applied to a 40-unit generator for PSO WTEED problem**

| PD[MW] | **10500** | **11000** | **12000** | **11300** |
|---|---|---|---|---|
| $P_1$[MW] | 114.0000 | 36 | 114 | 114 |
| P2[MW] | 114.0000 | 114 | 114 | 114 |
| P3[MW] | 120.0000 | 120 | 60 | 120 |
| P4[MW] | 190.0000 | 190 | 80 | 190 |
| P5[MW] | 97.0000 | 97 | 97 | 97 |
| P6[MW] | 96.6707 | 68 | 68 | 68 |
| P7[MW] | 223.6840 | 300 | 300 | 300 |
| P8[MW] | 300.0000 | 300 | 300 | 300 |
| P9[MW] | 135.0000 | 300 | 135 | 300 |
| P10[MW] | 300.0000 | 300 | 130 | 300 |
| P11[MW] | 375.0000 | 375 | 375 | 375 |
| P12[MW] | 375.0000 | 375 | 94 | 375 |
| P13[MW] | 500.0000 | 500 | 500 | 500 |
| P14[MW] | 125.0000 | 500 | 500 | 500 |
| P15[MW] | 500.0000 | 125 | 500 | 500 |
| P16[MW] | 500.0000 | 500 | 500 | 500 |
| P17[MW] | 500.0000 | 500 | 500 | 500 |
| P18[MW] | 500.0000 | 500 | 500 | 500 |
| P19[MW] | 550.0000 | 550 | 550 | 550 |
| P20[MW] | 550.0000 | 242 | 242 | 242 |
| P21[MW] | 550.0000 | 550 | 550 | 550 |
| P22[MW] | 550.0000 | 550 | 254 | 550 |
| P23[MW] | 254.0000 | 550 | 550 | 550 |
| P24[MW] | 550.0000 | 254 | 550 | 550 |
| P25[MW] | 550.0000 | 550 | 550 | 550 |
| P26[MW] | 550.0000 | 550 | 550 | 550 |
| P27[MW] | 150.0000 | 150 | 150 | 150 |
| P28[MW] | 10.0000 | 150 | 150 | 150 |
| P29[MW] | 150.0000 | 150 | 150 | 150 |
| P30[MW] | 97.0000 | 97 | 47 | 97 |
| P31[MW] | 190.0000 | 190 | 60 | 190 |
| P32[MW] | 190.0000 | 60 | 190 | 190 |
| P33[MW] | 190.0000 | 190 | 190 | 190 |
| P34[MW] | 90.0000 | 200 | 200 | 200 |
| P35[MW] | 200.0000 | 90 | 200 | 90 |
| P36[MW] | 90.0000 | 90 | 200 | 200 |
| P37[MW] | 110.0000 | 110 | 110 | 110 |
| P38[MW] | 110.0000 | 110 | 110 | 110 |
| P39[MW] | 110.0000 | 25 | 110 | 110 |
| P40[MW] | 550.0000 | 550 | 550 | 550 |
| Pw[MW] | 150.249 | 150.249 | 150.249 | 150.249 |
| $F_C$[$/h] | 157665.001 | 166404.803 | 164009.405 | 177272.170 |
| $P_L$[MW] | 1056.597 | 949.111 | 1038.643 | 1210.910 |
| $E_T$[Kg/h] | 199232.044 | 238948.748 | 211643.033 | 252972.712 |
| CEED[Kg/h] | 356940.591 | - | - | - |

### 4.4.3.1 The IEEE 30 bus system for 40 units results, and discussion

This is a fourth-generation unit system used to evaluate the effectiveness of the Lagrange multiplier method compared to other optimization methods for minimizing fuel cost and atmospheric emissions. Table 4.6 below shows that the fuel costs for the MGSO, GSA, GAEPSO, and LMM (1100 MW) algorithms are 1.11%, 4.09%, 6.60%, and 0.02%, respectively, compared to the PSO (1100 MW) algorithm. The proposed algorithm is 1–3% superior to other algorithms for solving the WTEED problem. The resultant NOx emissions are 28.96%, 35.18%, 21.11%, and 0.02% compared to the proposed PSO algorithm. The PSO method has effectively reduced pollutant levels by 1–22% compared to other optimization algorithms.

**Table 4.6:** **PSO compared to the LM method and other optimization algorithms for the WTEED problem applied to a 40-unit system, PD = 10500[MW]**

| **Algorithm** | **MSGO** | **GSA** | **GAEPSO** | **Developed PSO 1100[MW] wind power** | **Developed** LMM **1100[MW] wind power** |
|---|---|---|---|---|---|
| **Reference** | **(Secui et al.,2025)** | **(Jiang and** | **Wang.,2015)** | | |
| $P_1$[MW] | 549.9997828 | 103.9422 | 104.8952 | 114.0000 | 141.5002 |
| P2[MW] | 549.9997818 | 103.9422 | 113.0724 | 113.9999 | 141.5002 |
| P3[MW] | 119.9988116 | 112.2758 | 91.9834 | 119.9997 | 135.7827 |
| P4[MW] | 179.7374211 | 97.2549 | 161.3754 | 189.8484 | 192.7574 |
| P5[MW] | 96.99883835 | 179.8187 | 96.8920 | 97.0000 | 124.5002 |
| P6[MW] | 105.4022758 | 95.0966 | 139.2597 | 71.9253 | 158.0745 |
| P7[MW] | 299.9999742 | 119.5940 | 291.8542 | 299.9897 | 312.5901 |
| P8[MW] | 285.7135868 | 263.8120 | 261.6582 | 256.9563 | 327.5002 |
| P9[MW] | 287.9519754 | 263.0950 | 298.5412 | 300.0000 | 327.5002 |
| P10[MW] | 204.8086989 | 269.5109 | 140.7686 | 130.0000 | 194.3838 |
| P11[MW] | 243.6012566 | 130.8497 | 102.6502 | 236.1939 | 241.8042 |
| P12[MW] | 243.6003501 | 102.1717 | 94.9653 | 203.5049 | 239.8264 |
| P13[MW] | 394.2851366 | 95.9356 | 138.3682 | 500.0000 | 316.7899 |
| P14[MW] | 394.2799802 | 129.7004 | 308.3268 | 171.2636 | 346.1246 |
| P15[MW] | 394.2841752 | 301.7727 | 302.8832 | 175.4570 | 344.8558 |
| P16[MW] | 484.0292251 | 301.7773 | 306.0256 | 201.6798 | 344.8558 |
| P17[MW] | 489.2692227 | 496.6855 | 453.9314 | 485.4771 | 490.1056 |
| P18[MW] | 399.5209877 | 490.4517 | 478.8826 | 499.9893 | 491.8484 |
| P19[MW] | 506.0067276 | 502.7192 | 521.2446 | 549.8790 | 533.9051 |
| P20[MW] | 421.5364187 | 510.7183 | 526.3789 | 549.9993 | 533.9064 |
| P21[MW] | 433.5530376 | 523.3236 | 519.7761 | 548.8774 | 577.5002 |
| P22[MW] | 514.2353651 | 524.7491 | 516.7892 | 549.9889 | 577.5002 |
| P23[MW] | 433.5329212 | 523.4055 | 516.3568 | 549.9876 | 577.5002 |
| P24[MW] | 433.5268561 | 522.6936 | 508.0922 | 426.8130 | 577.5002 |
| P25[MW] | 433.5342317 | 522.6783 | 521.0308 | 550.0000 | 577.5002 |
| P26[MW] | 433.7489257 | 518.1215 | 519.1626 | 550.0000 | 577.5002 |
| P27[MW] | 10.05540017 | 149.7369 | 102.2686 | 10.0000 | 41.4425 |
| P28[MW] | 10.24418652 | 131.3113 | 132.0367 | 10.0019 | 41.4425 |
| P29[MW] | 10.0022597 | 130.7194 | 137.3896 | 10.0024 | 41.4425 |
| P30[MW] | 89.80426078 | 92.7962 | 96.5634 | 69.3648 | 124.5002 |
| P31[MW] | 189.998884 | 185.4022 | 168.7821 | 189.9978 | 203.4342 |
| P32[MW] | 189.9955403 | 164.0935 | 176.2361 | 189.9999 | 203.4342 |
| P33[MW] | 189.9978565 | 168.3946 | 168.7295 | 189.9995 | 203.4342 |
| P34[MW] | 199.9994215 | 166.0944 | 129.6826 | 199.9999 | 117.5002 |
| P35[MW] | 199.9994151 | 104.4641 | 185.6648 | 200.0000 | 117.5002 |
| P36[MW] | 199.998299 | 167.3644 | 168.0922 | 199.9882 | 117.5002 |
| P37[MW] | 110.0000000 | 89.9918 | 96.2245 | 25.0030 | 137.5002 |
| P38[MW] | 109.9997758 | 102.7658 | 104.6853 | 93.5475 | 137.5002 |
| P39[MW] | 109.9962344 | 108.1541 | 105.2245 | 84.0954 | 137.5002 |
| P40[MW] | 509.5675741 | 532.2987 | 532.2893 | 549.9995 | 533.9051 |
| Pw[MW] | 1100 | 121.0666 | 151.9035 | 1100 | 1100 |
| $F_C$[$/h] | 126645.134 | 142675.36 | 158269.6617 | 121107.584 | 121022.329581 |
| %Deviation of Fc for PSO compared with literature | 1.11 | 4.09 | 6. | 2.97 Average | 0.02 |
| $P_L$[MW] | 962.815076 | - | - | 1065.454 | 1063.149649 |
| $E_T$[Kg/h] | 239155.73184 | 366190.13 | 156826.1213 | 63713.052 | 75841.336841 |
| %Deviation of $E_T$ for PSO in relation to the literature | 28.96 | 35.18 | 21.11 | 22.39 (Average) | 4.35 |

Figures 4.7 and 4.8 graphically represent the relationship between fuel expenditure [$/h], power demand [MW], and emissions for various optimisation strategies.

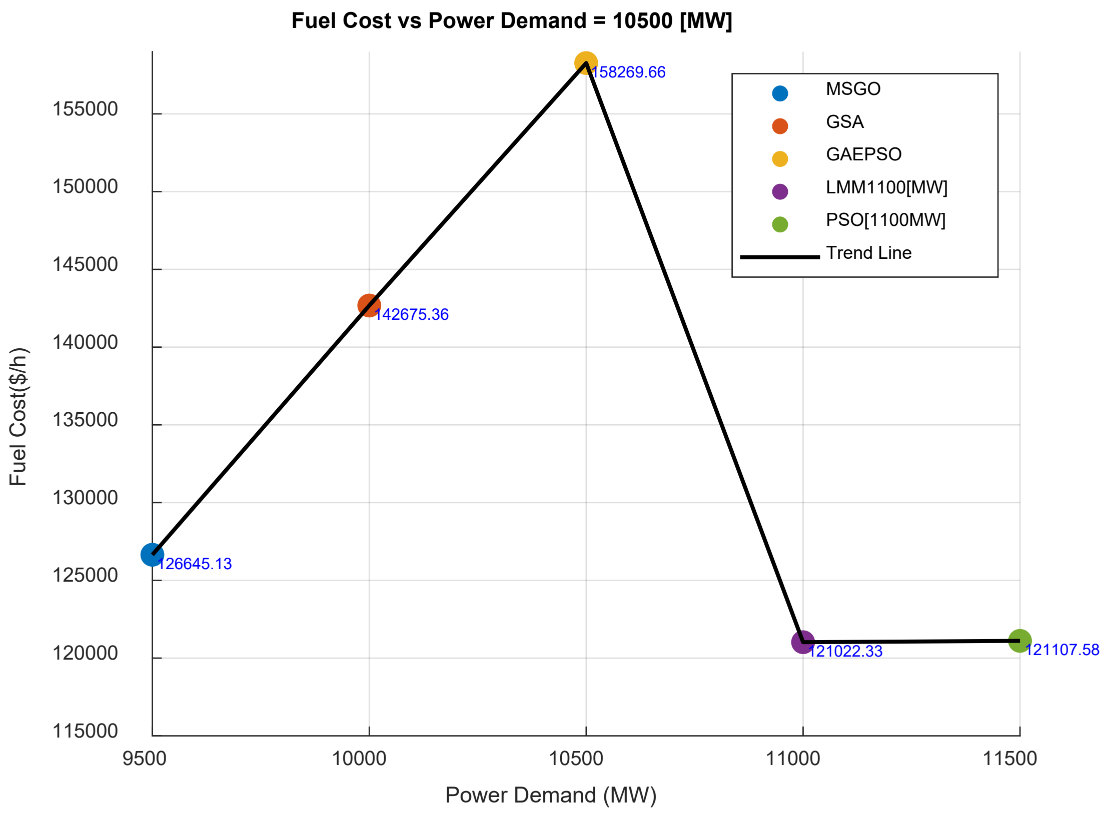



**Figure 4.7:** **Values of fuel cost and power demand for the WTEED problem using the PSO algorithm applied to a 40-unit system**

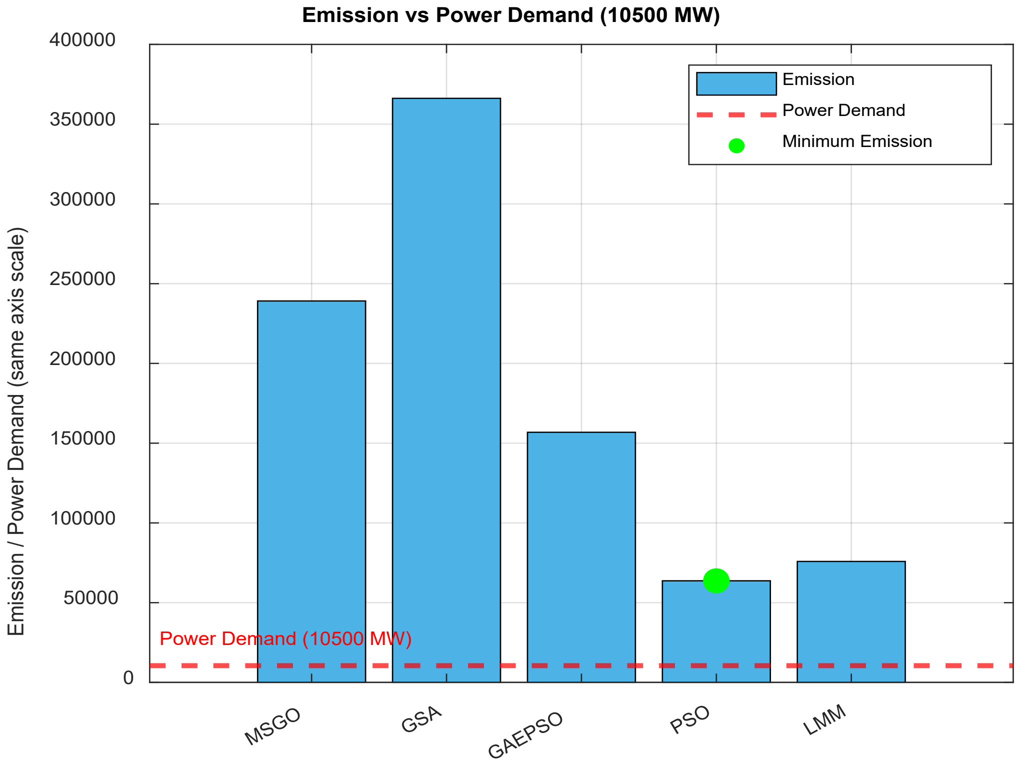



**Figure 4.8:** **Values of fuel cost and emission for the WTEED problem using the PSO algorithm applied to a 40-unit system**

## 4.5. Conclusion

In this chapter, a wind-thermal economic emission dispatch problem is successfully solved using the PSO method. Fuel and emission costs are minimised in the bi-objective problem of economic emission dispatch optimisation. This chapter presents the transmission line losses, emissions, and inequality limits. Three test systems, 6-unit, 10-unit, and 40-unit, are employed to evaluate the efficiency of the algorithm. Compared to other algorithms, the fuel cost and emissions for 6-unit systems are 1-3% and 1-2.8% lower, respectively. Additionally, compared with previous optimisation techniques, the 10-unit system's fuel cost and emissions are 1-3% and 1-2%, respectively. Lastly, fuel consumption and emissions for the bigger system have significantly decreased by 1-2% and 1-22%, respectively.

Lagrange's technique, which is described in Chapter 3, is compared with the developed PSO solution. Tables 4.2, 4.4, and 4.6 illustrate the fuel cost and emissions produced by the PSO algorithm compared with various alternative approaches. The number of particles in the swarm and the initial choice of the Lagrangian variable (lambda) determine the PSO and Lagrange’s solutions. It concludes that the PSO algorithm provides a near-global solution independent of the swarm size, whereas Lagrange’s method yields a global solution independent of the initial lambda.

It can be seen that the generation of random variables in each optimal solution has a significant impact on the solution to the WTEED problem across all test systems. The swarm population, iteration, solution weight, and constants c1 and c2 need to be carefully chosen. The 6-unit, 10-unit, and 40-unit models have the following computational times of 15.235862s, 7.024024s, and 10.978005s, respectively. In contrast, the best values for fuel cost, emissions, and total generation in the WTEED problem for the 6-unit, 10-unit, and 40-unit systems are within the expected values reported in the literature.

Based on the results described in Chapters 3 and 4, several calculations and simulations are required to thoroughly study the developed methods and their applications. The investigation will be further extended to a multi-area system in which the solution of the WTEED problem is distributed across subsystems, with a two-level hierarchical structure. In this decomposition, each area is treated as a separate problem. Since this can lead to high computational cost, future work beyond this thesis could use parallel computing within the MATLAB environment to solve the WTEED problem.

# CHAPTER FIVE

## MULTI-AREA WIND-THERMAL ECONOMIC EMISSION DISPATCH(MAWTEED) PROBLEM USING LAGRANGE MULTIPLIER METHOD(LMM)

### 5.1 Introduction

A Multi-Area Wind–Thermal Economic Emission Dispatch (MAWTEED) problem is addressed using the Lagrange multiplier approach. This optimisation is formulated as a multi-objective problem that reduces fuel cost and emissions. The inequality, quality limits, pollutants, transmission line loss, and tie-line power between areas are considered. To assess performance, the algorithm is tested on three benchmark systems: a 6-unit system with two areas, a 12-unit system with four areas, and a 40-unit system with two areas. For the 6-unit system, the fuel cost is 1-5% lower than that of other algorithms; for the 12-unit system, the fuel cost and emissions are 1-6% and 1-14% lower than those of other optimisation algorithms; and for the 40-unit system, there is a very significant reductions in fuel use by 1% and emissions by approximately 1-20%.

It appears that the Lagrange optimisation approach efficiently converges to an optimal solution by determining the best initial values of the lambda parameters across all test systems. In contrast, the optimum values for fuel cost, emissions, and total generation in the MAWTEED problem for the 6-unit, 12-unit, and 40-unit systems are within the predicted values published in the literature. Compared to heuristic methods documented in the literature, the conventional Lagrange employing decomposition coordination can easily produce optimal results for fuel and emission reduction. Lagrange's classical techniques are effective and precise for straightforward convex optimisation problems.

This chapter is further structured as follows. The problem formulation appears in Section 5.2; the development of Lagrange's decomposition coordinating method for solving the MAWTEED problem is discussed in Section 5.3, the algorithm of the Lagrange decomposition-coordinating method for solving the MAWTEED problem is presented in Section 5.4, various case studies used for multi-area WTEED problem solutions are provided in Section 5.5, and ultimately, the conclusion.

### 5.2 Multi-area wind-thermal economic emission dispatch problem formulation

The Multi-Area Wind-Thermal Dynamic Economic Dispatch (MAWTDEED) aims to minimise power transferred between generation areas, reduce emissions from thermal units, and control the power outputs of thermal and wind generating plants across multiple areas. This can be expressed mathematically as(Alli, 2024; Ghasemi et al., 2016; Hetzer et al., 2008; Ming Chen & Wang, 2010; Secui, 2015).

$$Min\ F_C = \sum_{m=1}^{M} \sum_{n=1}^{N_m} C_{mn}(P_{Gmn}) + C_{wmn}(P_{wmn}) \quad [\$/h] \quad (5.1)$$

Where, the number of online generators for the area $m$ in an $M$ area generation units denoted by $n$ is. $C_{wmn}(P_{wmn})$ is the amount of power that the wind turbine units produce.

The MAWTEED problem can be expressed mathematically as follows:

In order for equation 5.2 to provide the operation cost for thermal generators, find the vectors of the real power P and the power transmitted over the tie lines $P_T$.

$$F_C(P_{mn}) = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (a_{mn}P_{Gmn}^2 + b_{mn}P_{Gmn} + c_{mn}) \quad (5.2)$$

Where,

M Quantity of interconnected areas

$N_m$ The quantity of generators that are part of the producing power area in area m

$a_{mn}, b_{mn}, c_{mn}$ Price coefficient of the n^th generator in the m^nt area

$P_{Gmn}$ Real power formed by the n^th generator in the m^nt area

$P_{mn} = [P_{Gm1}P_{Gm2}P_{Gm3}P_{Gm4} \dots \dots P_{wm1}\ P_{wm2}\ P_{wm2}\ P_{wmN_m}]$ vector of an absolute power generated in the m^nth area and $m = \overline{1, M}$

$P = [P_1\ P_2\ P_3 \dots \dots P_M]^T$ Vector of the actual power generated throughout the entire power system

Equation (5.3) determines the operational cost for electricity being transmitted via the tie-lines.

$$F_T(P_T) = \sum_{m=1}^{M} \sum_{\substack{j=1 \\ j \neq m}}^{M} \left(q_{mj}P_{Tmj} + q_{jm}P_{Tjm}\right) \quad (5.3)$$

Where $q_{mj}$ is the transmission coefficient for the cost of power transmission from area m to area j, and $P_{Tmj}$ is the tie line power flow from area m to area j.

$P_{Tm} = \left[P_{Tm,m+1}\ \ P_{Tm,m+2}\ P_{Tm,m+3} \dots \dots P_{Tm,M}\right]^T$, $m = \overline{1, M-1}$ Power transfer vector between the m^th area and every other area.

$P = \left[P_{T1}\ \ P_{T2}\ P_{T3} \dots \dots P_{T,M-1}\right]^T$ Power transmission vector between all areas.

Equations (5.1), (5.2), and (5.3) are to be minimized, subject to the following limitations.

a) Lower and higher active power produced for every generator

$$P_{Gmn,min} \leq P_{Gmn} \leq P_{Gmn,max}, m = \overline{1, M}, \quad n = \overline{1, N_m}$$

$$0 \leq P_{wmn} \leq P_{wmn,rated}, m = \overline{1, M}, \quad n = \overline{1, N_m}$$

Where, $P_{Gmn,min}$ $and$ $P_{Gmn,max}$ The lowest and highest power that can be produced by the n^th generator in the m^th area. $P_{wmn,rated}$ is the rated wind power produced by the wind power system.

b) Lowest and highest active power sent through the tie lines.

$P_{Tmj,min} \leq P_{Tmj} \leq P_{Tmj,max}$. These limits apply to both power flow directions and can be expressed as follows:

$$P_{Tmin} \leq P_{Tmj} \leq P_{Tmax}, m = \overline{1, M}, \quad j = \overline{1, M}, j \neq m, \tag{5.4}$$

$$P_{Tmin} \leq P_{Tjm} \leq P_{Tmax}, m = \overline{1, M}, \quad j = \overline{1, M}, j \neq m$$

Power balance constraints.

Equation (5.5) below provides the balance between production and power demand for the m^th area and the whole system.

$$\sum_{n=1}^{N_m} P_{mn} = P_{Dmn} + P_{Lmn} + \sum_{\substack{m=1 \\ m \neq k}}^{M} \left[P_{Tmj} - (1 - \rho_{jm}) \times P_{Tjm}\right], m = \overline{1, M}, \tag{5.5}$$

The output power from both thermal and wind generators is represented in this instance by $P_{mn}$. Equation (5.6) provides the wind output power.

$$W_{j,k}(P_{w_{j,k}}) = \phi_{D_{j,k}} P_{E_{j,k}} + \phi_{OE_{j,k}} \times (P_{A_{j,k}} - P_{E_{j,k}}) + \phi_{UE_{j,k}} \times (P_{E_{j,k}} - P_{A_{j,k}}) \quad [\$/h] \tag{5.6}$$

Where

$P_{E_{j,k}}$ Estimated wind power output of the turbine $j$ in a wind farm $k$ [MW]

$P_{A_{j,k}}$ Accessible wind power output of the turbine $j$ in wind farm $k$ [MW]

$\phi_{D_{j,k}}$ Direct cost coefficient of the turbine $j$ in wind farm $k$ [$/MWh]

$\phi_{OE_{j,k}}$ Overestimation cost coefficient of the turbine $j$ in wind farm $k$ [$/MWh]

$\phi_{UE_{j,k}}$ Underestimation cost coefficient of the turbine $j$in wind farm $k$ [$/MWh]

Where,

$$P_{Lm} = \sum_{n=1}^{N_m} \sum_{r=1}^{N_m} (P_{mn} B_{mnr} P_{mr}) + \sum_{n=1}^{N_m} (B_{0n} P_{mn} + B_{m00}), m = \overline{1, M}, \tag{5.7}$$

$$\sum_{n=1}^{N_m} P_{mn} = P_{Dm} + \sum_{n=1}^{N_m} \sum_{r=1}^{N_m} (P_{mn} B_{mnr} P_{mr}) + \sum_{n=1}^{N_m} (B_{0n} P_{mn} + B_{m00}), + \tag{5.8}$$

$$+ \sum_{\substack{m=1 \\ m \neq k}}^{M} \left[P_{Tmj} - (1 - \rho_{jm}) \times P_{Tjm}\right], m = \overline{1, M},$$

Where, $B_{mnr}$, $B_{0n}, B_{m00}$, are the integrated power system transmission loss coefficients., $\rho_{jm}$ is the fractional loss rate from area j to area m, $P_{Tmj}$ and $P_{Tjm}$ are a power flow from area m to area j and area j to area m. Pollutant minimisation is also considered. The mathematical expression can be given as follows.

$$F_e = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (d_{mn} P_{mn}^2 + e_{mn} P_{mn} + f_{mn}) \quad (5.9)$$

$d_{mn}, e_{mn}, f_{mn}$ Pollutants coefficients of the n[th] generator in the m[nt] area. Equation (5.9b) provides the multi-area combined economic emission dispatch fuel cost.

$$F_T = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (a_{mn} P_{mn}^2 + b_{mn} P_{mn} + c_{mn}) + h_{mn} (d_{mn} P_{mn}^2 + e_{mn} P_{mn} + f_{mn}) \quad (5.9b)$$

## 5.3 Solution of the MAWTEED problem using Lagrange's decomposition coordinating method.

This section presents a Lagrange decomposition-coordinating technique for solving the multi-area wind-thermal economic emission dispatch problem. The method's development is based on the Lagrange functional for the multi-area economic dispatch problem. The following describes how the Lagrange functional is constructed:

$$L = \Bigg\{ \sum_{n=1}^{M} \sum_{n=1}^{N_m} (a_{mn} P_{mn}^2 + b_{mn} P_{mn} + c_{mn}) + h_{mn} (d_{mn} P_{mn}^2 + e_{mn} P_{mn} + f_{mn}) + \sum_{m=1}^{M} \sum_{\substack{j=1 \\ j \neq m}}^{M} \left( q_{mj} P_{Tmj} + q_{jm} P_{Tjm} \right) + \sum_{m=1}^{M} \Bigg\{ \lambda_m \Bigg( - \sum_{n=1}^{N_m} P_{mn} + P_{Dm} + \sum_{n=1}^{N_m} \sum_{r=1}^{N_m} (P_{mn} B_{mnr} P_{mr}) + \sum_{n=1}^{N_m} (B_{0n} P_{mn} + B_{m00}), + \sum_{\substack{m=1 \\ m \neq j}}^{M} \left[ P_{Tmj} - (1 - \rho_{jm}) \times P_{Tjm} \right] \Bigg) \Bigg\} \Bigg\} \quad (5.10)$$

Where $\lambda = (\lambda_1, \lambda_2, \lambda_3 \ldots . \lambda_M)$, are vectors of Lagrange multipliers, and $h_{mn}$ is the price penalty factor for every area. It is essential to determine the values of $P_{mn}, P_{Tmj}, P_{Tjm}$ *and* $\lambda$ such that the Lagrange functional has an optimum value according to $P_{mn}, P_{Tmj}, P_{Tjm}$, and the maximum according to $\lambda$ under the following limits.

$$P_{mn,min} \leq P_{mn} \leq P_{mn,max}, m = \overline{1, M}, \quad n = \overline{1, N_m} \quad (5.11)$$

$$0 \leq P_{wmn} \leq P_{wmn,rated}, m = \overline{1, M}, \quad n = \overline{1, N_m} \quad (5.12)$$

$$P_{Tmin} \leq P_{Tmj} \leq P_{Tmax}, m = \overline{1, M}, \quad j = \overline{1, M}, j \neq m, \quad (5.13)$$

$$P_{Tmin} \leq P_{Tjm} \leq P_{Tmax}, m = \overline{1, M}, \quad j = \overline{1, M}, j \neq m \quad (5.14)$$

The following requirements for optimality are employed to determine the optimal solution:

$$\frac{\partial L}{\partial P_{mn}} = 2a_{mn} P_{mn} + b_{mn} + h_{mn} (2d_{mn} P_{mn} + e_{mn}) + \lambda_m \left( 2 \sum_{r=1}^{N_m} B_{mnr} P_{mr} + B_{m0n} - 1 \right) = 0 = m = \overline{1, M}, n = \overline{1, N_m} \quad (5.15)$$

$$\frac{\partial L}{\partial P_{Tmj}} = q_{mj} + \lambda_m = 0 = e_{pTmj}, m = \overline{1, M}, \quad j = \overline{1, M}, j \neq m \quad (5.16)$$

$$\frac{\partial L}{\partial P_{Tjm}} = q_{jm} - \lambda_m(1-\rho_{jm}) = 0 = e_{pTjm}, m = \overline{1,M}, \quad j = \overline{1,M}, j \neq m \tag{5.17}$$

$$\frac{\partial L}{\partial \lambda_m} = -\sum_{n=1}^{N_m} P_{mn} + P_{Dm} + \sum_{n=1}^{N_m}\sum_{r=1}^{N_m}(P_{mn}B_{mnr}P_{mr}) + \sum_{n=1}^{N_m}(B_{0n}P_{mn} + B_{m00}), + \sum_{\substack{m=1 \\ m\neq j}}^{M}\left[P_{Tmj} - (1-\rho_{jm}) \times P_{Tjm}\right], m = \overline{1,M}, \quad j = \overline{1,M}, j \neq m \tag{5.18}$$

The solution to the above system of equations (5.15-5.18) can be obtained in parallel using a decomposition method based on the selection of coordinating variables. These are selected to be $\lambda_{m,} m = \overline{1,M,}$ . In a two-level structure, a coordinator provides the following values for the coordinating variables:

$$\lambda_m = \lambda_m^l, m = \overline{1,M,} \tag{5.19}$$

Where $l$ is the index of iterations. Equations (5.16) through (5.18) must be solved on the second level of the calculation framework, but equation (5.15) can be solved on the first level. The following method can be used to solve equation (5.15).

$$\frac{\partial L}{\partial P_{mn}} = 2a_{mn}P_{mn} + b_{mn} + h_{mn}(2d_{mn}P_{mn} + e_{mn}) + \lambda_m\left(2\sum_{r=1}^{N_m} B_{mnr}P_{mr} + B_{m0n} - 1\right) = 0, m = \overline{1,M}, n = \overline{1,N_m} \tag{5.20}$$

Solution of equation 5.20 by multiplying by ${}^{1}/_{2\lambda_m}$ be

$$\left(\frac{a_{mn}+h_{mn}d_{mn}}{\lambda_m}\right)P_{mn} + \sum_{r=1}^{N_m} B_{mnr}P_{mr} = \frac{1}{2}\left(1 - \frac{b_{mn}+h_{mn}e_{mn}}{\lambda_m} - B_{m0n}\right),$$

$$m = \overline{1,M}, n = \overline{1,N_m} \tag{5.21}$$

This can also be expressed in matrix form as:

$$\begin{bmatrix} \frac{a_{m1}+h_{m1}d_{m1}}{\lambda_m} + B_{m11} & B_{m12} & B_{mN_m1} \\ B_{m21} & \frac{a_{m2}+h_{m2}d_{m2}}{\lambda_m} + B_{m22} & B_{m2N_m} \\ B_{mN_m1} & B_{mN_m1} & \frac{a_{mN_m}+h_m d_m}{\lambda_m} + B_{mm} \end{bmatrix} \begin{bmatrix} P_{m1} \\ P_{m2} \\ P_{m3} \\ \vdots \\ P_{mN_m} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \left[1 - \left(\frac{b_{m1}+h_{m1}e_{m1}}{\lambda_m}\right) - B_{m01}\right] \\ \left[1 - \left(\frac{b_{m1}+h_{m1}e_{m1}}{\lambda_m}\right) - B_{m02}\right] \\ \left[1 - \left(\frac{b_{m1}+h_{m1}e_{m1}}{\lambda_m}\right) - B_{m0N_m}\right] \end{bmatrix}, m = \overline{1,M} \tag{5.22}$$

The following straightforward matrix form can be used to express equation (5.22).

$$E_m P_m = D_m, m = \overline{1,M} \tag{5.23}$$

With a known value of $\lambda_{m,} m = \overline{1,M,}$ .The solution for the active power of the m$^{th}$ area is given by

$$P_m = E_m \backslash D_m \tag{5.24}$$

The solutions of equations (5.16) and (5.17) can be found using the gradient procedure in the following way:

a) Initial values of $P_{Tmj}^{q_m} and\ P_{Tjm}^{q_m}$ are estimated, $q_m = I, q_m = \overline{I, k_m}$
$P_{Tmj} = P_{Tmj}^{q_m}$ and $P_{Tjm} = P_{Tjm}^{q_m}$, where $q_m$ is the index of the gradient process in the m^th area.

(b) The amended values of the tie lines' power are:

$P_{Tmj}^{q_m+1} = P_{Tmj}^{q_m} - \alpha_{Tmj} e_{PTmj}^{q_m}$, where $e_{PTmj}^{q_m}$ is given in equation (5.16), $m = \overline{1, M}$ (5.25)
$P_{Tjm}^{q_m+1} = P_{Tjm}^{q_m} - \alpha_{Tjm} e_{PTjm}^{q_m}$ where $e_{PTjm}^{q_m}$ is given in equation (5.17), $m = \overline{1, M}$. (5.26)
The solution is checked against the error constraints.
$e_{PTmj}^{q_m} \leq \varepsilon_1$ and $e_{PTjm}^{q_m} \leq \varepsilon_2$ where $\varepsilon_1$ and $\varepsilon_2$, They are tiny positive numbers.

Solutions of equations (5.24), (5.25), and (5.26) depend on $\lambda_{m,} m = \overline{1, M,}$. When the value of lambda is attained, the optimal values of $P_{mn}, P_{Tmj} and\ P_{Tjm}$ are found.

A gradient technique is utilised to compute the optimal value of $\lambda_{m,} m = \overline{1, M,}$ using the following form:

$$\lambda_m^{l+1} = \lambda_m^l + \alpha_{m\lambda} e_{m\lambda}, m = \overline{1, M}, \tag{5.27}$$

Where the steps of the gradient procedure are represented by $\alpha_{m\lambda}$. The gradient on the second level halts when $e_\lambda \leq \varepsilon_3,\ \varepsilon_3 \geq 0.$ The two-level solution structure for MAWTEED is shown in Figure 5.1 below.

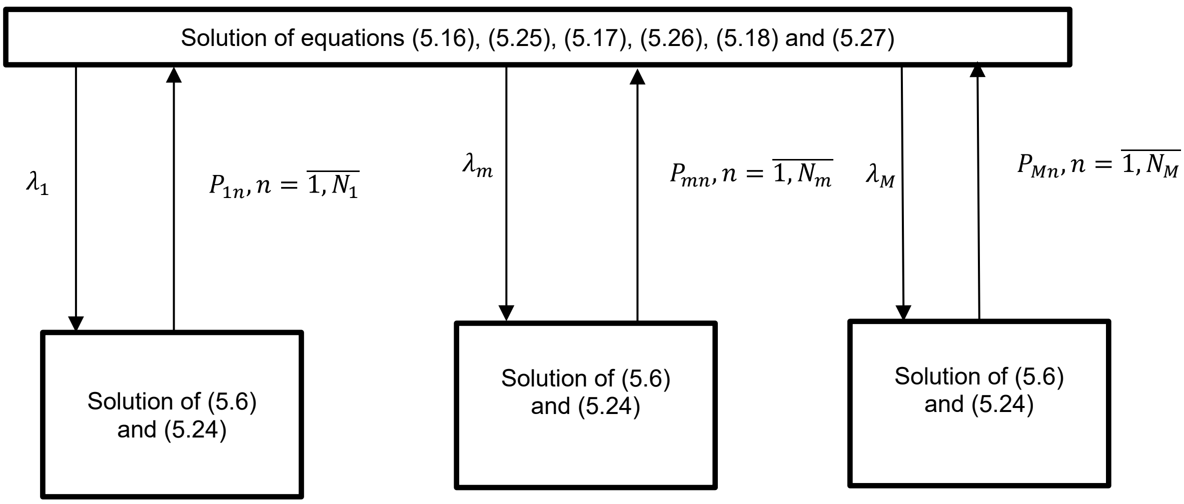



**Figure 5.1:** **Lagrange decomposition coordinating algorithm for the solution of the MAWTEED problem in two level structure**

## 5.4 Algorithm of the Lagrange decomposition-coordinating method for calculation of the MAWTEED problem.

**Step 1.** The number of iterations of the first and second levels is set, and all coefficient values are supplied $k_{m,} m = \overline{1, M,}$.

**Step 2**. Start values of the coordinating variables $\lambda_m^l, m = \overline{1, M,}$ are guessed.

**Step 3**. Beginning values of the tie lines are estimated. $P_{Tmj}^{k_m}, P_{Tjm}^{k_m}, m = \overline{1, M,} j \neq m$

**Step 4.** Computation on the first level is complete for every subsystem, $m = \overline{1, M,}$ using equation (5.24).

**Step 5**. The solution of the first level $P_{mn}^l, m = \overline{1, M,} n = \overline{1, N_m}$ are verified against the limits (5.11) and (5.12) and are sent to the second level.

**Step 6.** On the second level

(i) Equations (5.16) and (5.25)

(ii) Equations (5.17) and (5.26)

The gradient technique stops when this condition $e_{PTmj} \leq \varepsilon_1, and\ e_{PTjm} \leq \varepsilon_2$ is achieved or the highest number of iterations $k_{m,} m = \overline{1, M,}$ is attained.

The solution of the second level $P_{Tmj}^l and\ P_{Tjm}^l, m = \overline{1, M,} j = \overline{1, M,} j = m$ are verified under the conditions (5.13) and (5.14)

**Step 7**. Equations (5.18) and (5.27) are established. When the epsilon requirements are met, the gradient process ends. The related optimal solutions of the first-level problems are obtained along with the optimal value for $\lambda_{m,} m = \overline{1, M,}$. In the aforementioned approach, the optimal solution of the MAWTEED sub-problems in the m$^{th}$ region is obtained by performing the maximum number of iterations for one step of the gradient procedure for the optimization of $\lambda_{m,}$on the second level $k_{m,}$. Each area may have a distinct value for $k_{m,}$.

## 5.5 Studies of the multi-area MAWTEED problem solutions

Three benchmark models for multi-area power systems are used to test the proposed algorithm:

(i) Two areas with 6-unit generators and three thermal generating units in each area, including the transmission line losses, tie-line power transfer between areas, and wind power of 330MW injected in area 1.

(ii) Four areas with 3 generators in each area, including the transmission line losses, and 330MW of wind power from three wind farms supplying 110MW each.

(iii) Two areas with 20 generating units in each area, and a wind farm of 330MW, applied in area 1.

The three benchmark representations are implemented for the two considered scenarios:

(a) The entire power system is divided into multiple sections due to tie-line constraints.

Figure 5.2 illustrates the flowchart for Lagrange's decomposition-coordinating technique employed to determine the MAWTEED problem.

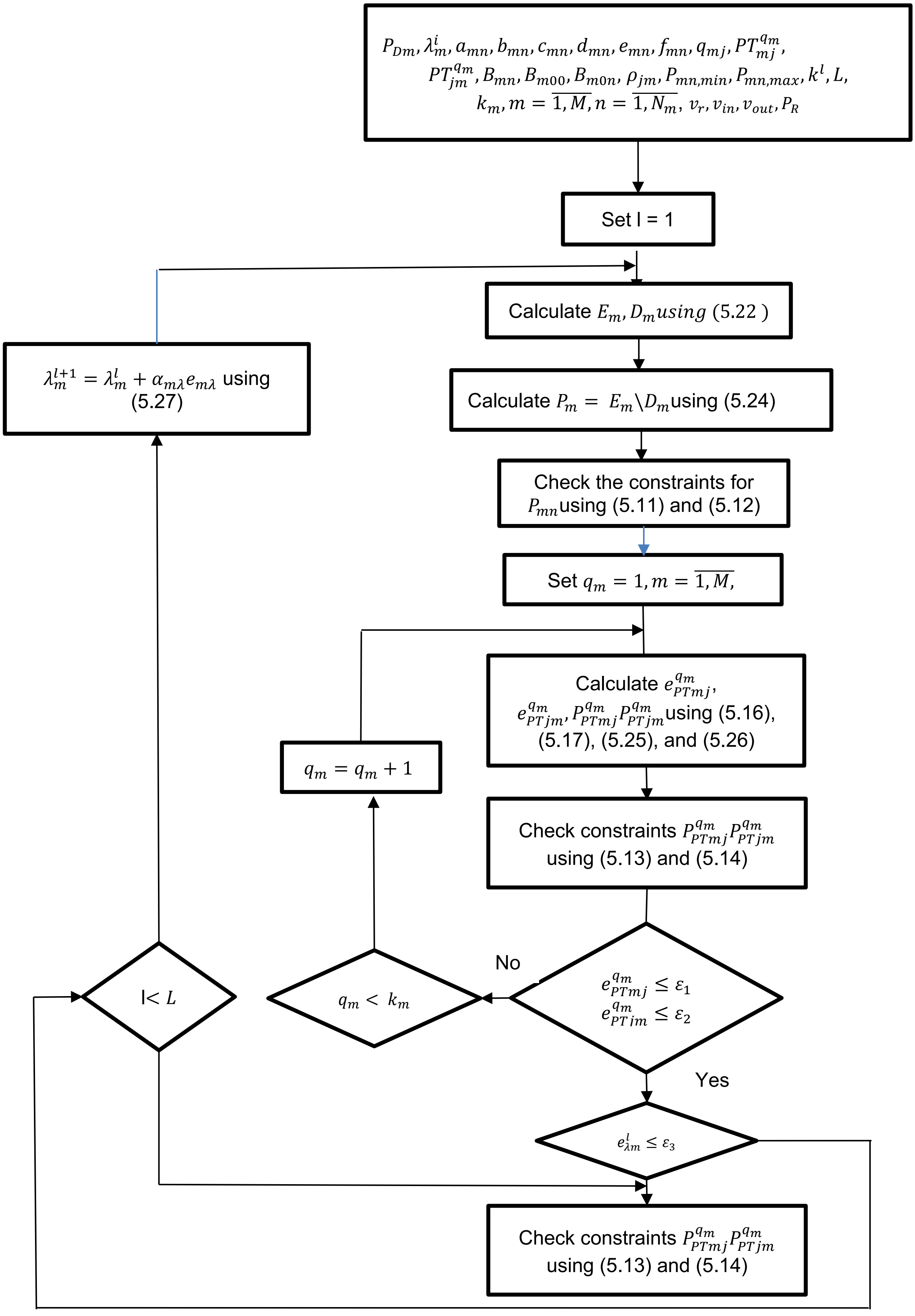



**Figure 5.2:** Lagrange's decomposition-coordinating algorithm flowchart for solving the MAWTEED problem

### 5.5.1 Test system I: IEEE30 bus system using 6-units

The test case consists of two areas with six thermal units, each with three generators. The Three-wind Park network that supplies wind power of 110MW each to the entire dispatch problem(Alli, 2024). The total power demand is 1263 [MW]. The area demand is 60% for area 1, and 40% for area 2. The demand for area 1 is 757.8 MW, and for area 2, 505.2MW. The data for the fuel cost is taken from (Basu, 2013) and B-loss coefficients (Krishnamurthy & Tzoneva, 2012). Various power demand values are presented in Table 5.1 to experiment with the system's behaviour to changes in power demand. The MATLAB script file is called MAWTEED_case6units.m, and the software is provided in Appendix C1.

**Table 5.1: The MAWTEED problem using the Lagrange multiplier method with various power demands**

| PD[MW] | 600 | 700 | 800 | 900 | 1000 | 1263 |
|---|---|---|---|---|---|---|
| P1,1[MW] | 50 | 50 | 100 | 160.7681 | 214.8556 | 232.7670 |
| P1,2[MW] | 50 | 50 | 50 | 97.5242 | 114.5397 | 120.1745 |
| P1,3[MW] | 50 | 50 | 50 | 81.2430 | 92.8587 | 96.7054 |
| P2,1[MW] | 91.8718 | 91.9027 | 143.2658 | 131.9209 | 131.9208 | 238.7650 |
| P2,2[MW] | 72.5071 | 72.5274 | 106.3055 | 98.8447 | 98.8447 | 169.1088 |
| P2,3[MW] | 63.0326 | 63.0426 | 79.7222 | 76.0380 | 76.0380 | 110.7344 |
| Pw[MW] | 292.4709 | 239.2799 | 304.3865 | 262.216 | 282.4045 | 313.9185 |
| PL[MW] | 3.6383 | 3.6393 | 5.7538 | 8.5456 | 11.4528 | 19.1629 |
| $F_C$[$/h] | 5256.8085 | 5257.3266 | 6129.7873 | 7047.359 | 7716.0091 | 9819.9287 |
| $E_T$[Kg/h] | 239.2494 | 239.2799 | 313.2204 | 378.9017 | 499.4681 | 830.2617 |
| CEED[Kg/h] | 7827.5305 | 7828.3065 | 9328.5035 | 10495.02 | 11702.766 | 16427.6875 |
| Computational time(s) | | | | | | |

The solution of the MAWTEED problem is shown in Table 5.1 above. The real power generators (P1,1-P1,3) and (P2,1-P2,3) units represent the thermal generators for area 1 and area 2, wind (Pw) generators represent the wind power generators for all three wind parks in [MW], fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h]. The obtained fuel cost using LMM is compared with Artificial Bee Colony Optimization (ABCO), Differential Evolution (DE), Evolutionary Programming (EP), Real-coded Genetic Algorithm (RCGA), and Semi-define Programming (SDP)(Basu, 2013), and (Alli, 2024) algorithms reported in the literature, as shown in Table 5.2 below.

**Table 5.2:** **The MAWTEED problem using the Lagrange multiplier method is compared with other optimisation algorithms using PD = 1263[MW]**

| Reference | (Basu.,2013) | | | | (Alli,2024) | Proposed |
|---|---|---|---|---|---|---|
| **Algorithm** | **ABCO** | **DE** | **EP** | **RCGA** | **SDP** | **LMM** |
| P1,1[MW] | 500.0000 | 500.0000 | 500.0000 | 500.0000 | 311.0838 | 232.7670 |
| P1,2[MW] | 200.0000 | 200.0000 | 200.0000 | 200.0000 | 200.0000 | 120.1745 |
| P1,3[MW] | 149.9997 | 150.0000 | 149.9919 | 149.6328 | 150.0000 | 96.7054 |
| P2,1[MW] | 204.3358 | 204.3341 | 206.4493 | 205.9398 | 162.8737 | 238.7650 |
| P2,2[MW] | 154.9954 | 154.7048 | 154.8892 | 155.8322 | 159.5473 | 169.1088 |
| P2,3[MW] | 67.2915 | 67.5770 | 65.2717 | 65.2209 | 101.7656 | 110.7344 |
| Pw [MW] | - | - | - | - | - | 313.9185 |
| $P_L$[MW] | 13.6224 | 13.6159 | 13.6021 | 13.6257 | - | 19.1629 |
| $F_C$[$/h] | 12255.39 | 12255.42 | 12255.43 | 12256.23 | 10722 | 9819.9287 |
| %Deviation of Fc for LMM in relation with the literature | 5.51625 | 5.51631 | 5.51633 | 5.51795 | 2.19568 | 4.8525 Average |
| $E_T$[Kg/h] | - | - | - | - | - | 830.2617 |
| CEED[Kg/h] | - | - | - | - | - | 16427.6875 |
| Computational time(s) | | | | | | |

Figure 5.3 represents the relationship between fuel cost [$/h] and power demand [MW] for various optimisation strategies, as described in Table 5.2 above.

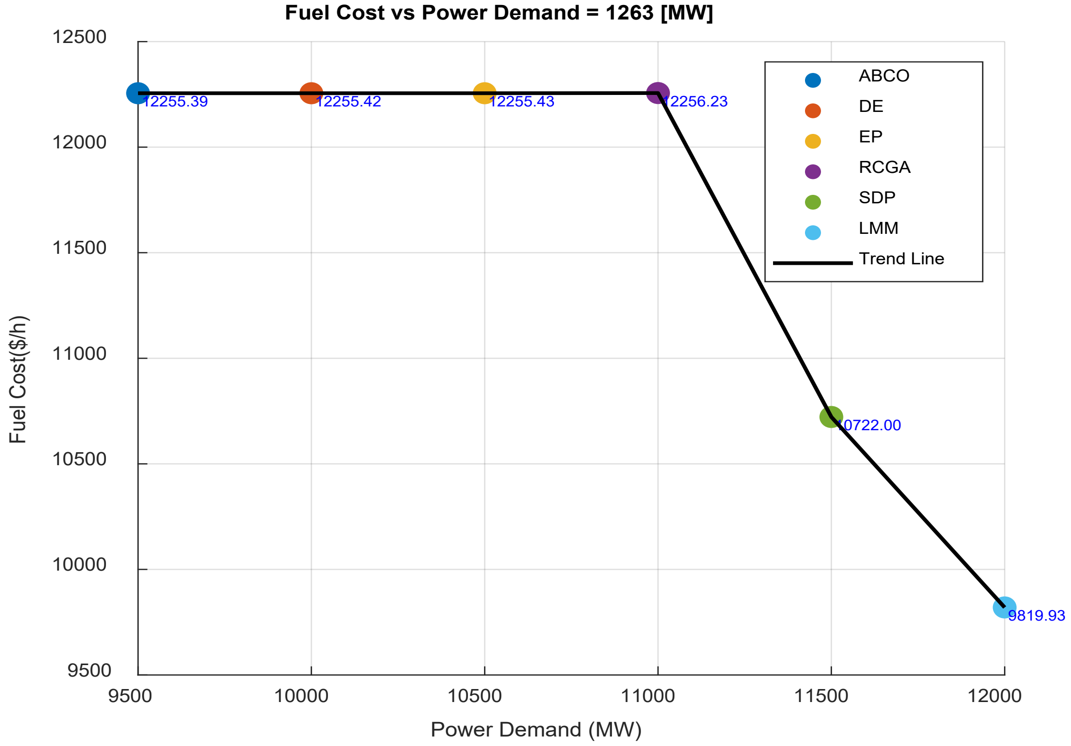



**Figure 5.3:** **Fuel cost versus power demand for different optimisation methods using the LM algorithm for a 6-unit system**

#### 5.5.1.1 The MAWTEED problem results and discussion for the 6-Unit System

This system utilises three wind farm networks, which supply 330 [MW] to the power dispatch problem. The following parameters are used for the Lagrange multiplier method. The initial value of lambda is [50 50], the total number of iterations m is 100, the tolerance value epsilon is 0.01, and the tolerance value of delta-lambda epsilon is 0.1. The proposed algorithm for a 6-unit multi-area system is compared with other optimization algorithms in the literature, namely, Artificial Bee Colony Optimization (ABCO), Differential Evolution (DE), Evolutionary Programming (EP), Real-coded Genetic Algorithm (RCGA), and Semi-define Programming (SDP)(Basu, 2013), and (Alli, 2024). The comparison is of the simulation results shown in Table 5.2 above.

The fuel cost results for the ABCO, DE, EP, RCGA, and SDP algorithms are 5.52%, 5.52%, 5.52%, 5.52%, and 2.19%, respectively, compared to the LMM algorithm. When using the LLM algorithm, there is a 1–5% benefit in fuel consumption when wind power is added to the economic emission dispatch as compared to other optimisation algorithms. The optimisation algorithm required only 57 iterations to find the optimal solution to the MAWTEED problem. The optimised value of lambda is [16.8244 24.5375], compared to the initial guess of [50 50].

### 5.5.2 Test System 2: IEEE30 bus system using 12-unit system

This test system consists of a standard IEEE 30-bus system with four areas, 12 thermal generators (3 in each area), and a three-wind-park network that supplies 110MW of wind power to the entire dispatch problem. The total power demand is 2090 [MW]. The area demand is considered to be 500 [MW] for area 1, 410 [MW] for area 2, 580 [MW] for area 3, and 600 [MW] for area 4.  The fuel cost and wind parameters are formed based on the ones stated in (Krishnamurthy et al., 2017) and (Alli, 2024). In Table 5.3, varying power demand is used to test the system's behaviour. The MATLAB script file is called MAWTEED_case12units.m, and the software is accessible in Appendix C2.

**Table 5.3: The MAWTEED problem using the LM Method with several power demands for 12-unit generators.**

| PD[MW] | 1500 | 1600 | 1700 | 1800 | 1900 | 2090 |
|---|---|---|---|---|---|---|
| P1,1[MW] | 37.7323 | 35 | 89.1836 | 112.8288 | 139.8413 | 87.5974 |
| P1,2[MW] | 130.0000 | 130 | 141.0986 | 179.4117 | 223.0174 | 138.5237 |
| P1,3[MW] | 125.0000 | 125 | 135.5624 | 172.9403 | 215.4290 | 133.0488 |
| P2,1[MW] | 84.8628 | 84.5174 | 80.6747 | 81.0847 | 79.5809 | 86.1913 |
| P2,2[MW] | 109.2794 | 108.8269 | 103.7930 | 104.3300 | 102.3600 | 110.0000 |
| P2,3[MW] | 125.0000 | 125.0000 | 125.0000 | 125.0000 | 125.0000 | 125.0000 |
| P3,1[MW] | 90.3138 | 90.3498 | 87.1977 | 88.6285 | 86.1077 | 146.1331 |
| P3,2[MW] | 125.3639 | 125.4012 | 122.1359 | 123.6182 | 121.0065 | 182.9771 |
| P3,3[MW] | 106.9614 | 107.0043 | 103.2550 | 104.9568 | 101.9586 | 173.4331 |
| P4,1[MW] | 65.4137 | 94.2511 | 90.0618 | 91.0632 | 88.7304 | 141.6098 |
| P4,2[MW] | 73.5199 | 104.8905 | 100.3432 | 101.4305 | 98.8973 | 156.0605 |
| P4,3[MW] | 114.5042 | 171.0612 | 162.8729 | 164.8311 | 160.2685 | 262.9685 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Pw[MW] | 299.0776 | 317.9222 | 325.9862 | 309.0305 | 307.4669 | 329.2499 |
| PL[MW] | 14.3517 | 17.8147 | 18.7403 | 22.7079 | 27.4143 | 33.6248 |
| $F_C$[$/h] | 69451.7517 | 76514.9109 | 77470.6989 | 82461.9824 | 86268.2387 | 106977.016 |
| $E_T$[Kg/h] | 587.7043 | 646.6456 | 650.3996 | 748.8189 | 877.4164 | 1120.2618 |
| CEED[Kg/h] | 106859.1545 | 119055.4491 | 119408.165 | 128990.227 | 138137.2521 | 181987.099 |

The solution of the MAWTEED problem is shown in Table 5.3 above. The real power generators (P1,1-P1,3), (P2,1-P2,3), (P3,1-P3,3), and (P4,1-P4,3) units represent the thermal generators for area 1, area 2, area 3, and area 4. Wind (Pw) generators represent the wind power generators for all three wind parks in [MW], fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h]. The obtained fuel cost and emission using LMM is compared with Lagrange's Decomposition-Coordinating Method (LDCM), Particle Swarm Optimisation (PSO), and Simplex-Particle Swarm Optimisation (SPSO)(Chopra et al., 2018) Algorithms found in the literature are shown in Table 5.4 below.

**Table 5.4:** **Lagrange multiplier method compared with other optimisation algorithms for MAWTEED problem, PD = 2090[MW]**

| Algorithm | LDCM | PSO | SPSO | LMM (Krishnamurthy,2011) | Proposed LMM |
|---|---|---|---|---|---|
| | (Chopra et al., 2018) | | | | |
| P1,1[MW] | 131.45 | 160.9 | 165 | 182.7879 | 87.5974 |
| P1,2[MW] | 209.49 | 163.3 | 260 | 302.6497 | 138.5237 |
| P1,3[MW] | 202.25 | 289.4 | 265 | 287.4964 | 133.0488 |
| P2,1[MW] | 150 | 109.6 | 150 | 130.5891 | 86.1913 |
| P2,2[MW] | 110 | 117.1 | 75 | 110.0000 | 110.0000 |
| P2,3[MW] | 191.63 | 184.4 | 175 | 147.6606 | 125.0000 |
| P3,1[MW] | 175 | 171.5 | 175 | 142.9309 | 146.1331 |
| P3,2[MW] | 215 | 197.5 | 160 | 181.2219 | 182.9771 |
| P3,3[MW] | 236.38 | 198.3 | 280 | 168.9267 | 173.4331 |
| P4,1[MW] | 164.09 | 230 | 175 | 126.3394 | 141.6098 |
| P4,2[MW] | 180.2 | 144.8 | 160 | 140.9621 | 156.0605 |
| P4,3[MW] | 306.18 | 211.5 | 280 | 236.9631 | 262.9685 |
| Pw[MW] | - | - | - | - | 329.2499 |
| Fc[$] | 144058.2 | 136598.3 | 134557.6 | 125883.5243 | 106977.016 |
| %Deviation of Fc for LMM in relation with the literature | 7.38565 | 6.08052 | 5.70945 | 4.05962 | 5.80881 |
| PL[MW] | 61.82 | 48.22 | 43.804 | 68.5286 | 33.6248 |
| ET[T] | 1923.7 | 3713.93 | 1473.935 | 1684.6603 | 1120.2618 |
| %Deviation of $E_T$ for LMM in relation to the literature | 13.1972 | 26.8263 | 6.81662 | 10.0609 | 14.2253 |
| CEED[$/h] | 266400.92 | 277258.2 | 232500.67 | 225665.1721 | 181987.099 |
| %Deviation of CEED for LMM in relation with the literature | 9.41303 | 10.3726 | 6.09349 | 5.35727 | 7.8091 |

The relationships between fuel cost [$/h], Emission [Kg/h], and CEED with power demand [MW] are shown graphically in Figures 5.4, 5.5, and 5.6 for different optimisation algorithms.

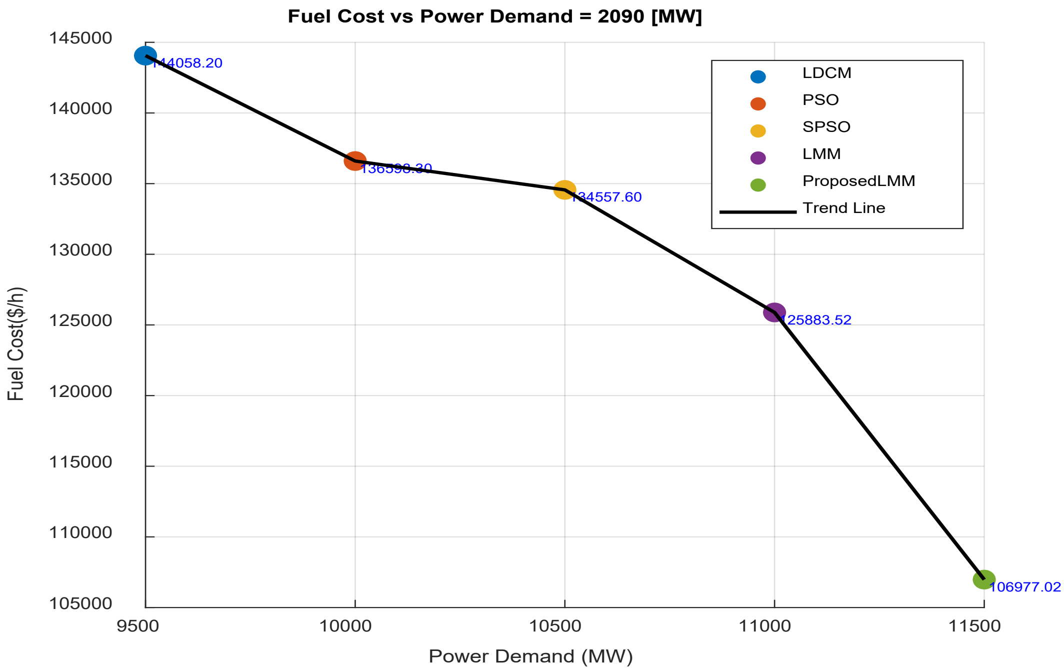



**Figure 5.4:** Fuel cost versus power demand for different optimisation methods using the LM algorithm for a 12-unit system

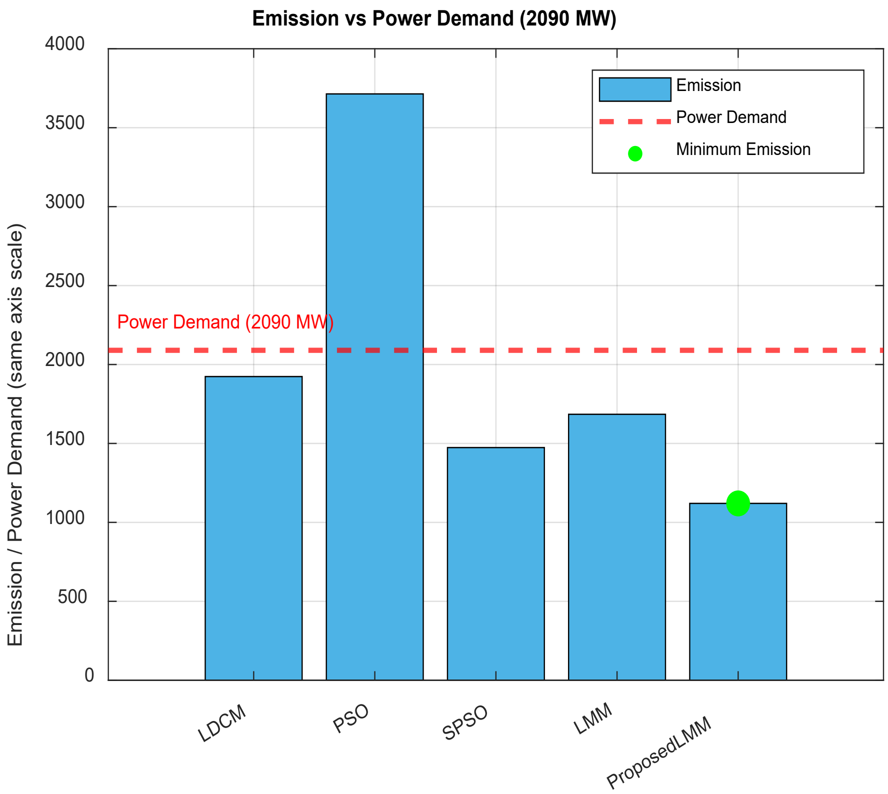



**Figure 5.5:** Emission versus power demand for different optimisation methods using the LM algorithm for a 12-unit system

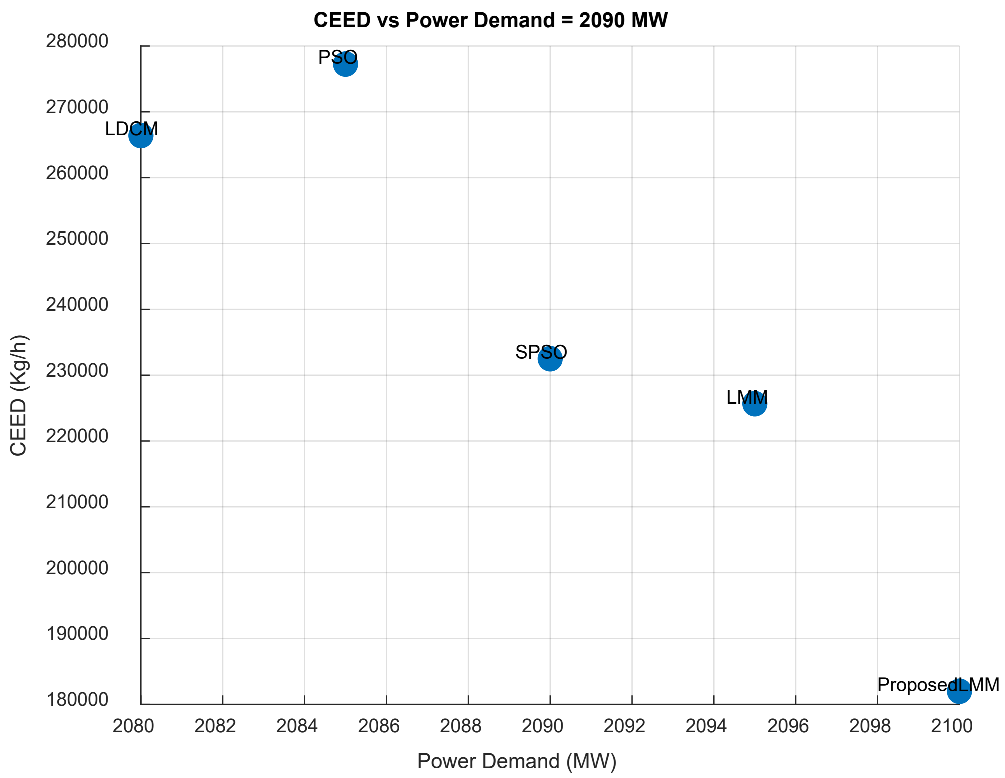



**Figure 5.6:** **CEED versus power demand for different optimisation methods using the LM algorithm for a 12-unit system**

### 5.5.2.1 The MAWTEED problem results, and discussion for the 12-Unit System

The Lagrange method is implemented to solve the multi-area economic emission dispatch problem with wind power, utilizing three wind-farm networks of 110MW each. The following parameters are used for the Lagrange multiplier method. The initial value of lambda is [100 100 100 100], the total number of iterations m is 3000, the tolerance value (epsilon) is 0.01, and the tolerance value of delta-lambda epsilon is 0.1. The Lagrange multiplier method is compared with Lagrange's Decomposition-Coordinating Method (LDCM), Particle Swarm Optimisation (PSO), and Simplex-Particle Swarm Optimisation (SPSO)(Chopra et al., 2018). The fuel cost results for the LDCM, PSO, SPSO, and LMM algorithms are 7.39%, 6.08%, 5.71%, and 4.06%, respectively, compared to the proposed LMM algorithm. When using the LLM algorithm, there is a 1–6% benefit in fuel consumption when wind power is added to the economic emission dispatch as compared to other optimisation algorithms. The resultant NOx emissions are 13.197%, 26.83%, 6.82%, and 10.06% lower than those of the proposed LM algorithm. The LM method has effectively reduced pollutant levels by 1–14% compared to other optimization algorithms.

The optimization algorithm took 87 iterations to find the optimal solution of the MAWTEED problem. The optimized lambda values are [79.1408, 105.8244, 192.1400, 212.6355], compared to the initial guess of [100, 100, 100, 100]. The proposed algorithm demonstrates reduced fuel costs and lower emissions, proving to be more effective than all other optimisation algorithms.

### 5.5.3 Test System 3: IEEE30 bus system using 40 generating units

The MAWTEED problem is solved using the Lagrange algorithm on a standard IEEE 30-bus two-area system with 40 thermal generators (20 in each area) and a three-wind park network supplying 110MW each to the entire dispatch problem. The total power demand is considered as 10500MW. The area demand is considered as 7500 MW for area 1 and 3000 MW for area 2, respectively. The fuel cost and wind parameters are formed based on the ones reported in (Krishnamurthy & Tzoneva, 2016)and (Alli, 2024) and are shown in Tables 5.5 and 5.6.

**Table 5.5: Wind-speed data**

| $v_{in}$ Cut-in wind speed | $v_r$ Rated wind-speed | $v_o$ Cut-out wind speed | $w_r$ Rated Power | $C_{pw}$ Underestimation | $C_{rw}$ Overestimation | $d_w$ direct cost |
|---|---|---|---|---|---|---|
| 5 | 15 | 45. | 330 | 5 | 5 | 0 |

**Table 5.6: IEEE 30 Bus System data for 40 thermal systems**

| Generating Unit [MW] | Generator Limits | | Fuel cost coefficients | | | Emission coefficients | | |
|---|---|---|---|---|---|---|---|---|
| | $P_{min}$ | $P_{max}$ | $a_i$ [$/(MWh)$^2$] | $b_i$ [$/MWh] | $c_i$ [$/h] | $\alpha_i$ [t/(MWh)$^2$] | $\beta_i$ [t/MWh] | $\gamma_i$ [t/h] |
| 1 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 2 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 3 | 60 | 120 | 0.02028 | 7.07 | 309.54 | 0.0762 | −2.36 | 100 |
| 4 | 80 | 190 | 0.00942 | 8.18 | 369.03 | 0.054 | −3.14 | 120 |
| 5 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 6 | 68 | 140 | 0.01142 | 8.05 | 222.33 | 0.0854 | −3.08 | 80 |
| 7 | 110 | 300 | 0.00357 | 8.03 | 287.71 | 0.0242 | −3.06 | 100 |
| 8 | 135 | 300 | 0.00492 | 6.99 | 391.98 | 0.031 | −2.32 | 130 |
| 9 | 135 | 300 | 0.00573 | 6.6 | 455.76 | 0.0335 | −2.11 | 150 |
| 10 | 130 | 300 | 0.00605 | 12.9 | 722.82 | 0.425 | −4.34 | 280 |
| 11 | 94 | 375 | 0.00515 | 12.9 | 635.2 | 0.0322 | −4.34 | 220 |
| 12 | 94 | 375 | 0.00569 | 12.8 | 654.69 | 0.0338 | −4.28 | 225 |
| 13 | 125 | 500 | 0.00421 | 12.5 | 913.4 | 0.0296 | −4.18 | 300 |
| 14 | 125 | 500 | 0.00752 | 8.84 | 1760.4 | 0.0512 | −3.34 | 520 |
| 15 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 16 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 17 | 220 | 500 | 0.00313 | 7.97 | 647.85 | 0.0151 | −2.68 | 220 |
| 18 | 220 | 500 | 0.00313 | 7.95 | 649.69 | 0.0151 | −2.66 | 222 |
| 19 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |
| 20 | 242 | 550 | 0.00313 | 7.97 | 647.81 | 0.0151 | −2.68 | 220 |
| 21 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 290 |
| 22 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 285 |
| 23 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 24 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 25 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 26 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 27 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 28 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 29 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 30 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 31 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 32 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 33 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 34 | 90 | 200 | 0.0001 | 8.95 | 107.87 | 0.0012 | −3.48 | 65 |
| 35 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 36 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 37 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 38 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 39 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 40 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |

Varying power demand is used to test the system's behaviour, as shown in Table 5.7. The software is in Appendix C3, with the MATLAB script file named MAWTEED_case40units.m. The real power generators (P1,1-P1,20) units represent the thermal generators for area 1, generators from (P2,1-2,20) represent generators for area 2, and wind (Pw) generators represent wind power generators, all in [MW], fuel cost ($F_C$), total emission ($E_T$) in [kg/h], and CEED in [Kg/h].

**Table 5.7: The MAWTEED problem using the Lagrange multiplier method with various power demands for a 40-unit system**

| PD[MW] | **8000** | **8500** | **9000** | **9500** | **10000** | **10500** |
|---|---|---|---|---|---|---|
| P1,1[MW] | 62.6342 | 111.4095 | 93.4401 | 111.8273 | 114.0000 | 114.0000 |
| P1,2[MW] | 62.6342 | 111.4095 | 93.4401 | 111.8273 | 114.0000 | 114.0000 |
| P1,3[MW] | 60 | 83.8286 | 69.4501 | 84.1629 | 88.8032 | 120.0000 |
| P1,4[MW] | 58.1447 | 127.9018 | 105.9376 | 128.4125 | 135.5010 | 184.4384 |
| P1,5[MW] | 68 | 97.0000 | 89.7009 | 97.0000 | 97.0000 | 97.0000 |
| P1,6[MW] | 128.6335 | 100.0267 | 82.0652 | 100.4443 | 106.2409 | 140.0000 |
| P1,7[MW] | 135 | 224.8212 | 189.3844 | 225.6452 | 237.0816 | 300.0000 |
| P1,8[MW] | 135 | 251.8268 | 207.4993 | 252.8575 | 267.1632 | 300.0000 |
| P1,9[MW] | 130 | 255.1525 | 209.8292 | 256.2063 | 270.8334 | 300.0000 |
| P1,10[MW] | 94 | 130.0000 | 130.0000 | 130.0000 | 130.0000 | 194.2785 |
| P1,11[MW] | 94 | 164.3456 | 134.9710 | 165.0286 | 174.5085 | 239.9567 |
| P1,12[MW] | 125 | 161.8368 | 132.1500 | 162.5270 | 172.1078 | 238.2515 |
| P1,13[MW] | 125 | 216.7120 | 174.0377 | 217.7042 | 231.4763 | 326.5568 |
| P1,14[MW] | 125 | 238.2467 | 190.9862 | 239.3455 | 254.5978 | 359.8966 |
| P1,15[MW] | 125 | 238.5932 | 191.8553 | 239.6799 | 254.7634 | 358.8979 |
| P1,16[MW] | 220 | 238.5932 | 191.8553 | 239.6799 | 254.7634 | 358.8979 |
| P1,17[MW] | 220 | 360.8711 | 301.0534 | 362.2620 | 381.5668 | 500.0000 |
| P1,18[MW] | 242 | 362.0568 | 301.9115 | 363.4552 | 382.8657 | 500.0000 |
| P1,19[MW] | 242 | 392.7521 | 325.9265 | 394.3058 | 415.8723 | 550.0000 |
| P1,20[MW] | 62.6342 | 392.7530 | 325.9272 | 394.3068 | 415.8732 | 550.0000 |
| P2,1[MW] | 550 | 420.5658 | 550.0000 | 547.5998 | 550.0000 | 477.1028 |
| P2,2[MW] | 550 | 420.1264 | 550.0000 | 546.9982 | 550.0000 | 476.5912 |
| P2,3[MW] | 550 | 422.2360 | 550.0000 | 548.1981 | 550.0000 | 478.2959 |
| P2,4[MW] | 550 | 422.2360 | 550.0000 | 548.1981 | 550.0000 | 478.2959 |
| P2,5[MW] | 550 | 400.6600 | 550.0000 | 518.2773 | 550.0000 | 453.0060 |
| P2,6[MW] | 550 | 400.6600 | 550.0000 | 518.2773 | 550.0000 | 453.0060 |
| P2,7[MW] | 15.4979 | 10.0000 | 15.3898 | 12.6284 | 28.4012 | 11.0158 |
| P2,8[MW] | 15.4979 | 10.0000 | 15.3898 | 12.6284 | 28.4012 | 11.0158 |
| P2,9[MW] | 15.4979 | 10.0000 | 15.3898 | 12.6284 | 28.4012 | 11.0158 |
| P2,10[MW] | 97 | 96.9774 | 97.0000 | 97.0000 | 97.0000 | 97.0000 |
| P2,11[MW] | 187.3504 | 420.5658 | 186.5573 | 166.2908 | 190.0000 | 477.1028 |
| P2,12[MW] | 187.3504 | 420.1264 | 186.5573 | 166.2908 | 190.0000 | 476.5912 |
| P2,13[MW] | 187.3504 | 422.2360 | 186.5573 | 166.2908 | 190.0000 | 478.2959 |
| P2,14[MW] | 90 | 422.2360 | 90.0000 | 90.0000 | 90.0000 | 478.2959 |
| P2,15[MW] | 90 | 400.6600 | 90.0000 | 90.0000 | 90.0000 | 453.0060 |
| P2,16[MW] | 90 | 400.6600 | 90.0000 | 90.0000 | 90.0000 | 453.0060 |
| P2,17[MW] | 110 | 10.0000 | 110.0000 | 110.0000 | 110.0000 | 11.0158 |
| P2,18[MW] | 110 | 10.0000 | 110.0000 | 110.0000 | 110.0000 | 11.0158 |
| P2,19[MW] | 110 | 10.0000 | 110.0000 | 110.0000 | 110.0000 | 11.0158 |
| P2,20[MW] | 550 | 96.9774 | 550.0000 | 458.4336 | 550.0000 | 97.0000 |
| Pw[MW] | 312.3995 | 312.4521 | 305.7285 | 303.5730 | 298.7680 | 282.5545 |
| $F_C$[$/h] | 90408.3823 | 96712.8847 | 100631.4254 | 106292.5497 | 112214.6744 | 120069.1486 |
| $P_L$[MW] | - | - | - | - | - | - |
| $E_T$[Kg/h] | 42120.6871 | 43703.4949 | 49528.3791 | 54644.3179 | 63073.0950 | 79614.2253 |
| CEED[Kg/h] | 150011.2334 | 152808.9588 | 170151.9779 | 180787.3657 | 197097.3539 | 214795.2613 |

This study presents a comparison of the wind-thermal multi-area economic emission dispatch problem using the Lagrange algorithm in relation to other optimisation algorithms., Salp Swarm Algorithm(SSA), Flower Pollination Algorithm(FPA), and Backtracking Search Algorithm(BSA) (Alli, 2024), and Hybridizing Sum-Local Search Optimize (HSLSO)(Chaudhary et al., 2020), and are given in Table 5.8.

**Table 5.8:** **Lagrange algorithm compared with other optimization algorithms for MAWTEED applied to the IEEE 40-unit system for PD = 10500[MW]**

| **Algorithm** | **SSA** | **FPA** | **BSA** | **HSLSO** | **Developed** LMM **330[MW] wind power** |
|---|---|---|---|---|---|
| **Reference** | **(Alli.,2024)** | | | **(Chaudhary,2020)** | |
| P1,1[MW] | 113.9998 | 97.4000 | 97.3999 | 110.8012 | 114.0000 |
| P1,2[MW] | 113.9996 | 179.7331 | 179.7331 | 113.9997 | 114.0000 |
| P1,3[MW] | 120 | 87.7999 | 87.7999 | 120.0 | 120.0000 |
| P1,4[MW] | 179.73331 | 105.3999 | 105.3999 | 179.7331 | 184.4384 |
| P1,5[MW] | 96.0324 | 300 | 259.5996 | 95.551 | 97.0000 |
| P1,6[MW] | 140 | 284.5997 | 284.5996 | 140.0 | 140.0000 |
| P1,7[MW] | 300 | 295.3663 | 284.5997 | 300.0 | 300.0000 |
| P1,8[MW] | 284.5995 | 279.5997 | 204.7997 | 284.5997 | 300.0000 |
| P1,9[MW] | 284.6002 | 94 | 94 | 284.5997 | 300.0000 |
| P1,10[MW] | 269.9999 | 94 | 94 | 270.0 | 194.2785 |
| P1,11[MW] | 168.7999 | 214.7598 | 304.5195 | 94.0 | 239.9567 |
| P1,12[MW] | 350.0002 | 394.2794 | 394.2793 | 300.0 | 238.2515 |
| P1,13[MW] | 394.2794 | 394.2794 | 394.2793 | 304.5195 | 326.5568 |
| P1,14[MW] | 394.2793 | 394.2794 | 394.2793 | 394.2797 | 359.8966 |
| P1,15[MW] | 304.5197 | 489.2794 | 489.2794 | 484.0395 | 358.8979 |
| P1,16[MW] | 484.0391 | 399.5196 | 489.2793 | 484.0391 | 358.8979 |
| P1,17[MW] | 489.2794 | 511.2794 | 511.2794 | 489.2794 | 500.0000 |
| P1,18[MW] | 489.2796 | 511.2794 | 511.2793 | 489.2796 | 500.0000 |
| P1,19[MW] | 511.2794 | 511.2794 | 523.2794 | 549.9998 | 550.0000 |
| P1,20[MW] | 511.2793 | 511.2794 | 523.2794 | 511.2791 | 550.0000 |
| P2,1[MW] | 523.2795 | 523.2794 | 523.2794 | 523.2792 | 477.1028 |
| P2,2[MW] | 343.7598 | 523.2794 | 523.2793 | 523.2791 | 476.5912 |
| P2,3[MW] | 254 | 523.2794 | 523.2794 | 523.2794 | 478.2959 |
| P2,4[MW] | 523.2794 | 523.2794 | 523.2794 | 523.2794 | 478.2959 |
| P2,5[MW] | 523.2793 | 10 | 10 | 523.2795 | 453.0060 |
| P2,6[MW] | 523.2793 | 10 | 10 | 254.0 | 453.0060 |
| P2,7[MW] | 109.9999 | 10 | 10 | 10.0001 | 11.0158 |
| P2,8[MW] | 109.9999 | 87.7999 | 87.7999 | 10 | 11.0158 |
| P2,9[MW] | 110 | 190 | 190 | 10 | 11.0158 |
| P2,10[MW] | 87.7998 | 190 | 190 | 87.7997 | 97.0000 |
| P2,11[MW] | 159.733 | 190 | 190 | 188.5959 | 477.1028 |
| P2,12[MW] | 159.733 | 200 | 196.9619 | 159.7331 | 476.5912 |
| P2,13[MW] | 159.7331 | 164.7999 | 164.7998 | 159.733 | 478.2959 |
| P2,14[MW] | 90 | 164.7999 | 164.7998 | 164.8002 | 478.2959 |
| P2,15[MW] | 164.8 | 110 | 109.9997 | 164.7998 | 453.0060 |
| P2,16[MW] | 164.8 | 110 | 89.1141 | 164.7998 | 453.0060 |
| P2,17[MW] | 72.296 | 110 | 109.9997 | 89.1143 | 11.0158 |
| P2,18[MW] | 89.114 | 511.2794 | 511.2793 | 89.114 | 11.0158 |
| P2,19[MW] | 89.114 | 550 | 549.9992 | 89.1134 | 11.0158 |
| P2,20[MW] | 242 | 550 | 550 | 242.0001 | 97.0000 |
| Pw[MW] | - | - | - | - | 282.5545 |
| $F_C$[$/h] | 120857.2447 | 130573.2847 | 130580.7421 | 125100.2621 | 120069.1486 |
| %Deviation of Fc for LMM in relation with the literature | 0.16356 | 2.09544 | 2.09687 | 1.02605 | 1.34548 (average) |
| $P_L$[MW] | | | | | - |
| $E_T$[Kg/h] | | | | | 79614.2253 |
| CEED[Kg/h] | - | - | - | - | 214795.2613 |

The relationship between the fuel cost [$/h] and the power demand [MW] is shown graphically in Figure 5.7 for different optimization algorithms.

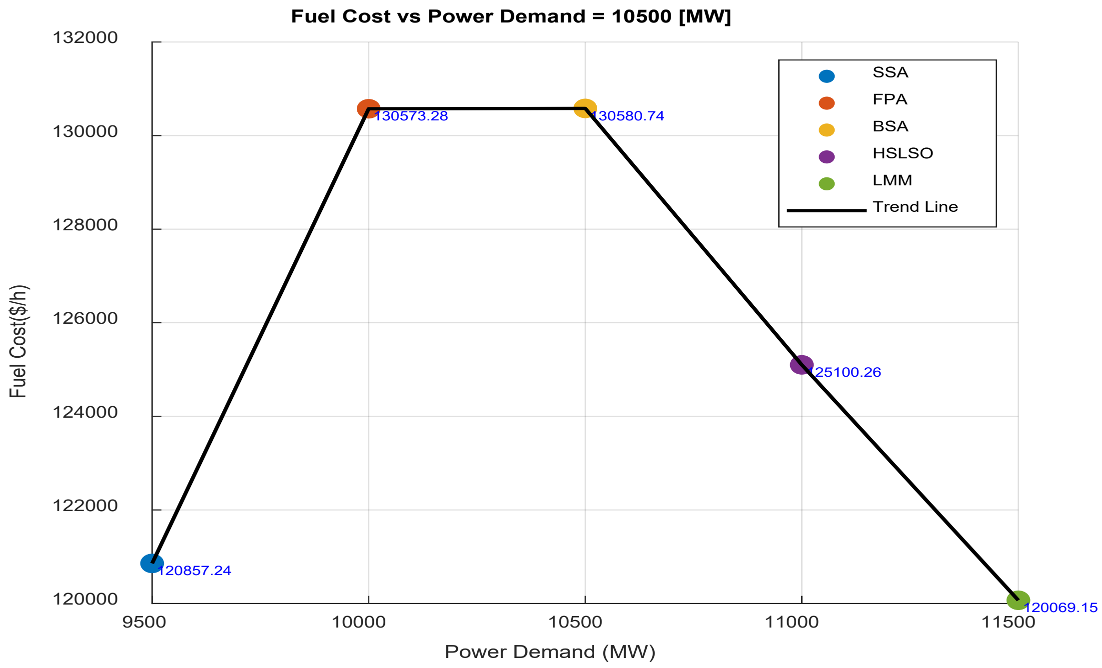



**Figure 5.7:** **Values of fuel cost and power demand for the MAWTEED problem using the LM algorithm for a 40-unit system**

As illustrated in Table 5.9, the fuel cost and emission values are compared with the statistical values of various algorithms.

**Table 5.9:** **LMM Comparison with various algorithms**(Ahmed et al., 2022)

| Algorithm | Fuel cost ($/h) | Fuel cost (%)Deviation | Emission (Kg/h) | Emission (%)Deviation |
|---|---|---|---|---|
| ABC | 126 480.56 | 1.30023 | 209 285.74 | 22.4423 |
| EMA | 125 910.69 | 1.1874 | 210 238.19 | 22.5328 |
| NSGA-II | 125 830.00 | 1.17138 | 210 950.00 | 22.6001 |
| SSA-WSA | 125 760.05 | 1.15749 | 206 705.97 | 22.194 |
| MOSSA | 125 591.29 | 1.12394 | 205 965.40 | 22.1219 |
| CSOA | 124 330.50 | 0.8718 | 116 560.5 | 9.41668 |
| **Developed LMM** | **120069.1486** | **1.13537** | **79614.2253** | **20.218** |

The best fuel [$/h] and emission [Kg/h] values for different optimization algorithms are compared with the LMM algorithm, as shown in Table 5.11 above. The best emission values are plotted against power demand[MW] as illustrated in Figure 5.8 below.

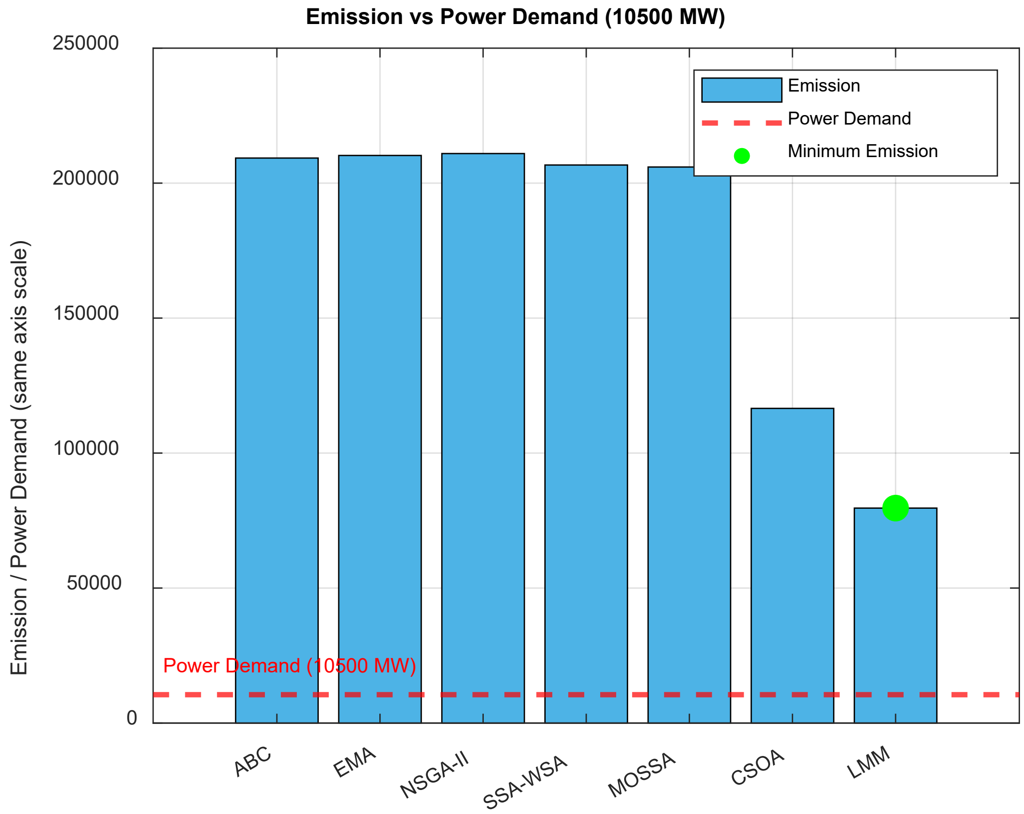



**Figure 5.8:** **Values of emission and power demand for the MAWTEED problem using the LM algorithm for a 40-unit system**

#### 5.5.3.1 The MAWTEED problem results, and discussion for a 40-Unit System

This is a fourth-generation unit system used to evaluate the effectiveness of the Lagrange multiplier method relative to other optimization methods for minimizing fuel costs and atmospheric emissions. The following parameters are used for the Lagrange multiplier method. The initial value of lambda is [ 50 50], the total number of iterations m is 10000, the tolerance value (epsilon) is 0.01, and the tolerance value of delta-lambda epsilon is 0.1. The Lagrange multiplier method is compared with different heuristic algorithms, i.e., Salp Swarm Algorithm(SSA), Flower Pollination Algorithm(FPA), and Backtracking Search Algorithm(BSA)(Alli, 2024), and Hybridizing Sum-Local Search Optimize (HSLSO)(Chaudhary et al., 2020). The comparison is made based on the best cost and lowest emissions. The summary of the comparison between the Lagrange method and the mentioned heuristic algorithms is given in Table 5.11.

The optimization algorithm took 3101 iterations to find the optimal solution of the MAWTEED problem. The calculated value of lambda after the optimization algorithm is [48.1302 22.3787], compared to the initial guess.

Table 5.11 shows that the values for fuel cost and emission minimization for LMM are 1.13537% and 20.218%, respectively, which are better than those for the MAWTEED problem.

## 5.6 Conclusions

In this chapter, a Multi-Area Wind–Thermal Economic Emission Dispatch problem is successfully solved using the Lagrange multiplier approach. Fuel and emission costs are minimised in the bi-objective problem of economic emission dispatch optimisation. This chapter presents the transmission line losses, emissions, and inequality limits. Three test systems, a 6-unit system with two areas, a 12-unit system with four areas, and a 40-unit system with two areas, are used to evaluate the algorithm. Compared to other methods, the fuel cost for 6-unit systems is reduced by 1–5%. Additionally, compared with previous optimisation techniques, the 12-unit system's fuel cost and emissions are 1–6% and 1–14%, respectively. Lastly, fuel consumption and emissions for the 40-unit system have decreased by 1% and 1–20%, respectively.

All test systems demonstrate that the Lagrange optimisation approach efficiently converges to an optimal solution by selecting the best initial lambda values. For fuel cost, emissions, and total generation in the MAWTEED problem, the optimal results for the 6-unit, 12-unit, and 40-unit systems are within the expected values reported in the literature. Compared to the heuristic methods described in the literature, the classical Lagrange approach utilising decomposition coordination can easily produce optimal results for fuel and emission reduction. For simple, convex cases, Lagrange's classical methods are reliable and accurate.

In chapter six, the Multi-Area Wind–Thermal Economic Emission Dispatch (MAWTEED) problem is solved using Particle Swarm Optimisation (PSO), and the MAWTEED dispatch problem solutions are compared between classical and heuristic optimisation techniques.

# CHAPTER SIX
# MULTI-AREA WIND-THERMAL ECONOMIC EMISSION DISPATCH(MAWTEED) PROBLEM USING PARTICLE SWARM OPTIMISATION(PSO)

## 6.1 Introduction

In this chapter, a Multi-Area Wind-Thermal Economic Emission Dispatch problem is solved using Particle Swarm Optimisation (PSO). Fuel and emission costs are minimised in the bi-objective problem of economic emission dispatch optimisation. This chapter presents the transmission line losses, emissions, and inequality limits. Three benchmark systems: a 6-unit system with two areas, a 12-unit system with four areas, and a 40-unit system with two areas, are used to test the PSO algorithm. Compared to other methods, the fuel cost for 6-unit systems is reduced by 1-5%. Additionally, compared with previous optimisation techniques, the 12-unit system's fuel cost and emissions are 1-6% and 1-12%, respectively. Finally, the 40-unit system shows a notable 1% reduction in fuel and a 14% reduction in emissions.

Efficient convergence of PSO is achieved through careful selection of algorithm parameters, including the inertia weight (w), cognitive coefficient (c1), social coefficient (c2), population size, and number of iterations. The fuel and emissions for the MAWTEED problem are affected by the random selection of PSO variables, and the resulting LMM and PSO values differ by 1% across all test systems. The optimum conditions for the 6-unit, 12-unit, and 40-unit systems, however, fall within the anticipated ranges for fuel cost, emissions, and total generation in the MAWTEED problem, as compared to the literature. Compared to the heuristic methods described in the literature, the classical Lagrange approach utilising decomposition coordination can easily produce optimal results for fuel and emission reduction. For simple, convex situations, Lagrange's classical methods are effective and accurate. On the other hand, when the objective function and constraints are nonlinear and nonconvex, the PSO algorithm can solve the MAWTEED problem more successfully than the Lagrange technique.

The chapter is further structured in the following sections. Section 6.2, Introduction to the PSO algorithm; Section 6.3, development of the PSO algorithm for the solution of the MAWTEED problem; Section 6.4, illustrates different case studies used for multi-area WTEED problem solutions.

## 6.2 Introduction to the PSO algorithm

Kennedy and Eberhart developed the particle swarm optimisation technique in 1995. It is an optimisation based on how the swarm's species behaves. It provides a population-based search strategy in which species, referred to as particles, navigate an open, multidimensional space and change their positions over time.

Based on both its own and the other particle's experiences, the particle positioned itself in the solution space, utilising the best position that both of them had encountered. Based on past experiences, the swarm follows the nearby particle. Every particle in the swarm travels around the search space at a specific speed that varies based on its prior experiences.

Let a particle coordinate (position) and its corresponding flight speed (velocity) in a search space denoted by $p$ and $v$ respectively. The $i^{th}$particle is preserved as a volume less particle assumed by $p_i = (p_{i1}, p_{i2}\dots p_{id})$ and the best position as ${p^{best}}_i = ({p^{best}}_{i1}, {p^{best}}_{i2}\dots {p^{best}}_{id})$.The index of the best particle amongst others is given a name called global best $g_d^{best}$ . The change of velocity of the $i^{th}$ particle is given by $v_i = (v_{i1}, v_{i2}\dots v_{id})$. Equations (6.1) and (6.2) provide the immediate velocity and the distance between ${p_{id}}^{best}$ and ${g_d}^{best}$, respectively, which may be utilised to determine the updated velocity and location of each particle.

$${v^{t+1}}_{id} = \omega {v^t}_i + c_1 rand()({p^{best}}_{id} - {p^t}_{id}) + c_2 rand()({p^{best}}_{id} - {p^t}_{id}) \quad (6.1)$$

$${{p_{id}}^{t+1}}_i = {p^t}_{id} + {v^{t+1}}_{id} \quad (6.2)$$

Where

$c_1$ Self-confidence ranges between 1.5 and 2, and

$c_2$ Swarm range between 2 and 2.5.

$({p^{best}}_{id} - {p^t}_{id})$ and $({p^{best}}_{id} - {p^t}_{id})$ are a particle memory influence and a swarm influence.

The proper balance between global and local search can be achieved by selecting the inertia weight factor $\omega$, to be between 0.9 and 0.4 during the search, and can be given by the following expression:

$$\omega = \omega_{\max} - \left[\frac{\omega_{\max} - \omega_{\min}}{iter_{\max}}\right] iter$$

Where,

| | |
|---|---|
| $\omega$ | The inertia weight factor |
| $\omega_{max}$ and $\omega_{min}$ | Maximum and minimum weighting factors |
| $Iter$ and $Iter_{max}$ | Current and maximum iterations. |

## 6.3 PSO algorithm developed for the solution of the MAWTEED problem

In Chapter 5, the Multi-Area Wind-Thermal Economic Emission Dispatch (MAWTEED) problem is formulated and solved using the Lagrange Multiplier Method(LMM). The problem is articulated using the total fuel cost equation for thermal and wind power systems as given in equation (5.1). The fuel cost for conventional and wind power generators is given by equations (5.2) and (5.6). This is subject to equality and inequality constraints given in equations (5.5), (5.7), (5.8), and (5.11-5.14). The wind power system is further analysed using the probability density function (PDF), yielding equations (2.46) and (2.51). The emission function is given by equation (5.9), and the price penalty factors (5.9b) are used to formulate the bio-objective problem into a single objective function for the MAWTEED problem. The MAWTEED problem is solved using a tailored PSO heuristic optimization algorithm. It is needed to associate the structure of the MAWTEED problem with that of the velocity and position Equations (6.1) and (6.2) adopted by the PSO algorithm. This can be achieved with the following steps:

In the wind thermal economic dispatch problem, the active power generated by conventional and wind power generators is represented by the positions of particle members; the number of generators is equal to the number of members in a particular particle in the swarm.

The velocity values are parameters that indicate the active power, even though they are used to search within the boundaries.

It is anticipated that the quantity of particles in the swarm is $N_P$. Equations (5.1) to (5.9b) provide the developed PSO algorithm for solving the multi-area combined economic emission dispatch problem:

**Step1:** Provide the initial values for the acceleration constants, inertia weight, and other PSO parameters. Considering c1 and c2, the uniform random variables are *rand1*, *rand2*, and the maximum number of iterations $Iter^{max}$.

**Step 2:** Compute the lowest and highest initial velocities using the generator limit restrictions in Equations (5.11) and (5.12), and are presented in Equation (6.3) as follows:

$$-0.5P_{mpi}^{min} \le V_{mpi} \le +0.5P_{mpi}^{max}, p = \overline{1, N_p}, i = \overline{1, n-1} \tag{6.3}$$

Where

$N_P$ Number of particles in the swarm

$n$ The total number of generators equals the number of members in a single particle.

The particle position and velocity are calculated for $(n-1)$ generators, since the first generator is acknowledged as a slack one.

**Step 3:** Compute the initial velocity of all members of the particles excluding the slack bus generator, using equation (6.4)

$$V_{mpi} = V_{mpi}^{min} + rand()(V_{mpi}^{max} - V_{mpi}^{min}), p = \overline{1, N_p}, i = \overline{1, n-1} \quad (6.4)$$

Where

$V_{mpi}^{min}, V_{mpi}^{max}$ Previously computed lowest and highest velocities

**Step 4:** Compute the initial Position of the particle members using Equation (6.5)

$$P_{mpi} = P_{mpi}^{min} + rand()(P_{mpi}^{max} - P_{mpi}^{min}), p = \overline{1, N_p}, i = \overline{1, n-1} \quad (6.5)$$

Wind power is added to equation (6.5) using equation (5.6) to obtain the active power of the MAWTEED problem in each area. Then this is verified if they are within the given limits in equation(5.12) and is represented by the following equation(6.6)

$$\begin{cases} P_{mpi}^{min}, & P_{mpi} \leq \quad P_{mpi}^{min} \\ P_{mpi}^{max}, & P_{mpi} \geq \quad P_{mpi}^{max} \\ P_{mpi}, & P_{mpi}^{min} \leq P_{mpi} \leq \quad P_{mpi}^{max} \end{cases}, p = \overline{1, N_p}, i = \overline{1, n-1} \quad (6.6)$$

The power system's buses are divided into three categories: load (PQ), generator (PV), and slack buses. Any bus connected to a generator with the greatest generating capacity is referred to as a slack bus in a power system. The system's voltage and angle are referenced by the slack bus. The slack bus's real and reactive powers are uncontrollable; the actual or reactive power necessary to balance the system's power flows is provided by the bus.

Generation scheduling is used to solve the economic dispatch problem without accounting for the actual power output of the slack bus generator. In the PSO algorithm, the slack bus serves to meet the power balance requirement specified in Equation (2.4). Step 5 outlines the process for determining the real power of the slack bus generator in the PSO algorithm.

**Step 5:** The slack bus generator is considered to have the most power-generating capability among dependent generators. Equation (6.7a) provides the power balance constraint that may be used to calculate the initial active power $P_{mpd}$ using the bus voltage magnitude and phase angle as a reference(Krishnamurthy & Tzoneva, 2012):

$$P_{mpd} + \sum_{\substack{i=1 \\ i \neq d}}^{n} P_{mpi} = \left\{ \sum_{\substack{i=1 \\ i \neq d}}^{n} \sum_{\substack{j=1 \\ j \neq d}}^{n} P_{mpi} B_{ij} P_{pj} + \sum_{\substack{j=1 \\ j \neq d}}^{n} P_{mpj} \left( B_{jd} + B_{dj} \right) P_{mpd} + B_{dd} P_{mpd}^2 + \sum_{\substack{i=1 \\ i \neq d}}^{n} B_{io} P_{mpi} + B_{do} P_{mpd} + B_{moo} + P_{Dm} \right\}, p = \overline{1, N_p}, \quad (6.7a)$$

Where

$P_{mpd}$ Is the power generated by the slack bus for each area.
$P_{Dm}$ Is the total power demand for each area

$\sum_{\substack{i=1\\i\neq d}}^{n} P_{mpi}$ Is the total active power of the power system, without the slack bus power, and is known by. $\sum_{\substack{i=1\\i\neq d}}^{n} P_{mpi} = \left[P_{mT1}, P_{mT2} \dots P_{mN_T}, P_{Wm1}, P_{Wm2} \dots P_{WN_w}\right]$, (6.7b)

Where,

$P_{TN_T}$ Is the active power generated by thermal power generators for the area.
$P_{WN_w}$ Is the active power generated by wind power generators for each area.

By using the quadratic form, Equation (6.7) can be rewritten, where the $P_{mpd}$ is the unknown variable in the equation.

$$XP_{mpd}^2 + YP_{mpd} + Z = 0 \quad (6.8)$$

Where

$$X = B_{dd} \quad (6.9)$$

$$\sum_{\substack{j=1\\j\neq d}}^{n} P_{mpj}\left(B_{jd} + B_{dj}\right) + B_{do} - 1 \quad (6.10)$$

$$Z = \sum_{\substack{i=1\\i\neq d}}^{n} \sum_{\substack{j=1\\j\neq d}}^{n} P_{mpi}B_{ij}P_{mpj} + \sum_{\substack{i=1\\i\neq d}}^{n} B_{io}P_{mpi} + B_{moo} + P_{Dm} - \sum_{\substack{i=1\\i\neq d}}^{n} P_{mpi} \quad (6.11)$$

Equation (6.8) positive root can be found as follows:

$$P_{mpd} = \frac{-Y+\sqrt{Y^2-4XZ}}{2X}, where\ Y^2 - 4XZ \geq 0 \quad (6.12)$$

Now the real power vector is shaped as the $P_{mp} = \left[P_{mpd}, P_{mpi}, i = 1, n, i \neq d\right]$

$where\ p = \overline{1, N_p}$

**Step 6:** Compute the following objective functions for the initial positions of the particles.

Fuel cost function

$$F_C(P_{mn}) = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (a_{mn}P_{Gmn}^2 + b_{mn}P_{Gmn} + c_{mn}) \quad (6.13)$$

Emission function

$$F_e = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (d_{mn}P_{mn}^2 + e_{mn}P_{mn} + f_{mn}) \quad (6.14)$$

Combined economic emission function

$$F_T = \sum_{n=1}^{M} \sum_{n=1}^{N_m} (a_{mn}P_{mn}^2 + b_{mn}P_{mn} + c_{mn}) + h_{mn}(d_{mn}P_{mn}^2 + e_{mn}P_{mn} + f_{mn}) \quad (6.15)$$

Where the Max-Max price penalty factor is computed as:

$$h_{mn} = \frac{a_{mn}P_{mn,max}^2 + b_{mn}P_{mn,max} + c_{mn}}{d_{mn}P_{mn,max}^2 + e_{mn}P_{mn,,max} + f_{mn}}, n = \overline{1, M, N_m} \quad (6.16)$$

The fuel cost parameters $F_C$ of all the particles are arranged in ascending order. The first position value in the ascending order is picked as the $F_T^{best}$

**Step 7:** Select the best initial position and the global best initial position as follows:

The initial particle locations in the swarm are assumed to be optimal.

$P_{mp}^{best} = minP_{mp}^{best}, p = \overline{1, N_p}, i = \overline{1, n}$ .

The best position out of all the best particles $minP_{mp}^{best},\ p = \overline{1, N_p}$ is taken as

$G_{mp}^{best} = minP_{mp}^{best}, p = \overline{1, N_p}$

$l^{th}$ step of the iteration process commences, where $l = l + 1$

**Step 8**: Calculate new velocity using Equation (6.17)

$$V_{pi}^{new^l} = \omega.V_{mpi}^{l-1} + c1 * rand1()\left(P_{mp}^{best^{l-1}} - P_{mpi}^{l-1}\right) + c1 * rand1()\left(G^{best^{l-1}} - P_{mpi}^{l-1}\right),$$

$$p = \overline{1, N_p}, i = \overline{1, n-1} \qquad (6.17)$$

Check the limits on the lowest and highest velocities.

$$\text{If } V_{mpi}^{new^l} > V_{mpi}^{max^{l-1}}, V_{mpi}^{new^l} = V_{mpi}^{max^{l-1}} \text{ and}$$

$$\text{If } V_{mpi}^{new^l} < V_{mpi}^{min^{l-1}}, V_{mpi}^{new^l} = V_{pi}^{min^{l-1}}, p = \overline{1, N_p}, i = \overline{1, n-1} \qquad (6.18)$$

**Step 9:** Apply Equation (6.18) to determine the generator's new locations within the particles.

$$\text{If } P_{mpi}^{new^l} = P_{mpi}^{l-1} + V_{mpi}^{new^l}, p = \overline{1, N_p}, i = \overline{1, n} \qquad (6.19)$$

**Step 10:** Employing the constraint Equation (6.6), determine the generators' new locations within the particles as follows:

$$P_{pi}^{new^l} = \begin{Bmatrix} P_{mpi}^{min}, & P_{mpi}^{new^l} \leq & P_{mpi}^{min} \\ P_{mpi}^{max}, & P_{mpi}^{new^l} \geq & P_{mpi}^{max} \\ P_{mpi}^{new}, & P_{mpi}^{min} \leq P_{mpi}^{new^l} \leq & P_{mpi}^{max} \end{Bmatrix}, p = \overline{1, N_p}, i = \overline{1, n-1} \qquad (6.20)$$

**Step 11:** Determine the new real power of the slack bus generator. Employing the formula from step 5, create the new real power vector $P_{mpi}^{new^l}$. Utilising the constraint Equation (6.6), determine the slack bus generators' new location within the particles.

$$P_{pd}^{new^l} = \begin{Bmatrix} P_{mpd}^{min}, & P_{mpd}^{new^l} \leq & P_{mpd}^{min} \\ P_{mpd}^{max}, & P_{mpd}^{new^l} \geq & P_{mpd}^{max} \\ P_{mpd}^{new}, & P_{mpd}^{min} \leq P_{mpd}^{new^l} \leq & P_{mpd}^{max} \end{Bmatrix} \qquad (6.21)$$

**Step 12:** The full active power vector for the $l^{th}$ iteration is:

$$P_{mpd}^{new^l} = \left(P_{mpd}^{new^l}, P_{mpi}^{new^l}, i \neq d, i = \overline{1,n}\right), p = \overline{1, N_p} \quad (6.22)$$

**Step 13:** Compute the new objective functions $F_{mT}^{new}$ using Step 6

**Step 14:** Verify the new objective function $F_T^{new}$ as explained below

$$\boldsymbol{If}\ F_{mT}^{new^l} < F_{mT}^{best^{l-1}} then\ F_{mT}^{best^l} = F_{mT}^{new^l} and\ P_{mpi}^{best^l} = P_{mpi}^{new^l}$$

Else $If\ F_{mT}^{best^l} = F_{mT}^{best^{l-1}} and\ P_{mpi}^{best^l} = P_{mpi}^{best^{l-1}}$ (6.23)

$$G^{best^l} = P_{mp}^{best^l}, p = \overline{1, N_p}$$

where $l$ is the number of iterations

The best solution $G^{best}$ is only one for the whole system. The best solution per particle is only one $P_{mp}^{best} = minP_{mpi}, i = \overline{1,n}$.

The best solution for the entire system is $P_{mp}^{best} = \left[P_{m1}^{best}, Pm_1^{best}, \dots., P_{N_p}^{best}\right]$. Then

$$G^{best} = minP_{mp}^{best^l}, p = \overline{1, N_p}$$

**Step 15:** Repeat steps 5-13 until the highest number of iterations is attained.

## 6.4 Case studies of the multi-area MAWTEED problem solutions

Three benchmark models for multi-area power systems are used to test the proposed algorithm:

(i) There are two areas with six generators and three thermal generating units each, after taking into account transmission line losses, tie-line power transfer across areas, and wind power of 330MW injected in area 1.

(ii) Four areas with three thermal units in each area, considering the transmission line losses, and 330MW of wind power from three wind farms supplying 110MW each.

(iii) Two areas with 20 generating units in each area, and a wind farm of 330MW, applied in area 1. The PSO flowchart is shown in Figure 6.1 below.

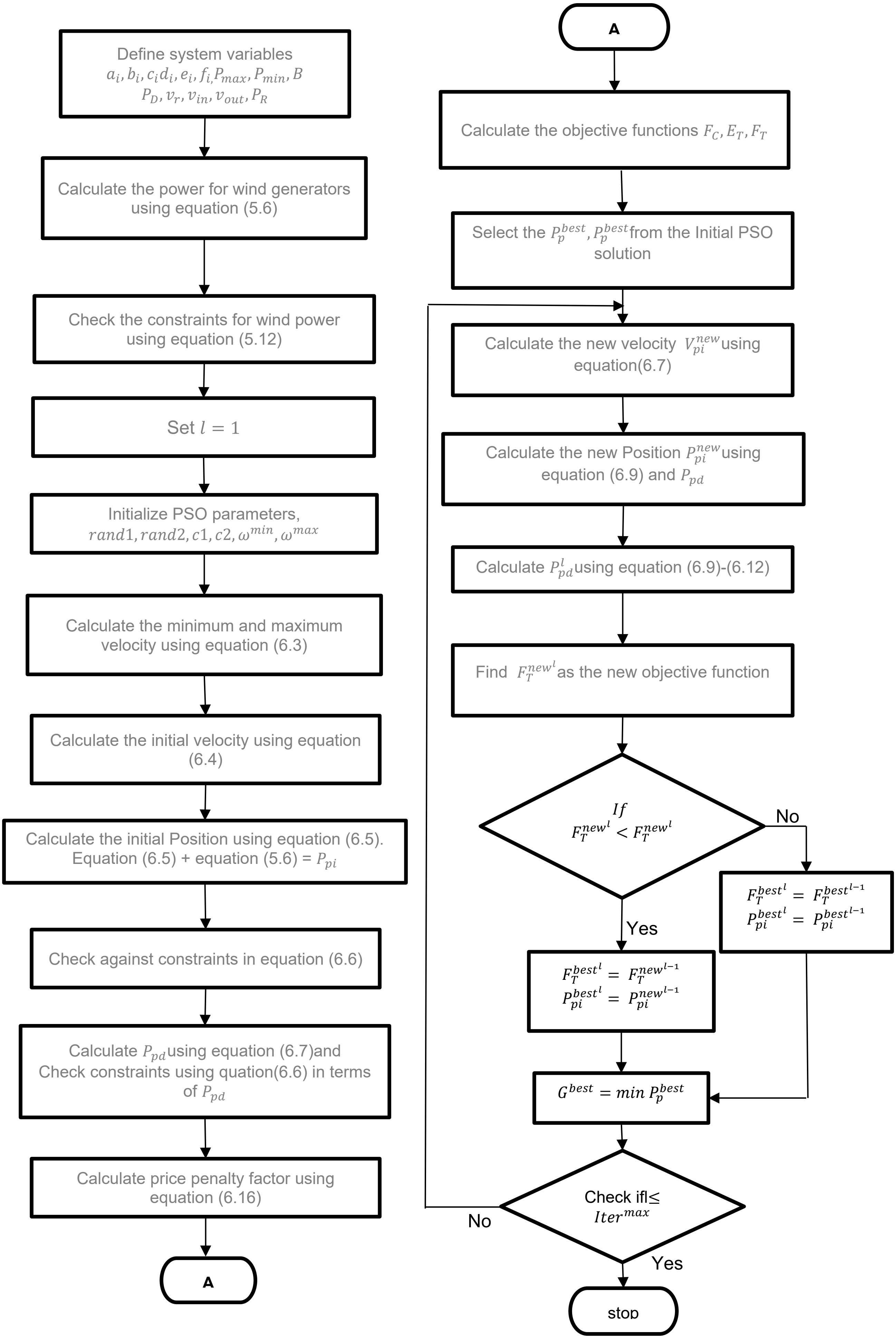



**Figure 6.1:** **The MAWTEED problem solution flowchart of applied to PSO algorithm**

### 6.4.1 Test system I: IEEE 30 bus system with 6 generating units.

The MAWTEED problem is solved using two areas with six conventional units, each containing three thermal generators. The three wind park network supplies 110MW of wind power each to the entire dispatch problem(Alli, 2024). The total power demand is 1263 [MW]. The area demand is 60% for area 1, and 40% for area 2. The demand for area 1 is 757.8 MW, and for area 2, it is 505.2MW. The data for the fuel cost is taken from (Basu, 2013)and B-loss coefficients(Krishnamurthy et al., 2017). Several power demand values are presented in Table 6.1 to validate the system's performance under fluctuations in power demand. The MATLAB script is called MAWTEED_casePSO6units. m, and the program is provided in Appendix D1.

**Table 6.1: The MAWTEED problem applied to the PSO algorithm with various power demands**

| PD[MW] | 600 | 700 | 800 | 900 | 1000 | 1263 |
|---|---|---|---|---|---|---|
| P1,1[MW] | 100 | 100.0000 | 130.83 | 102.306 | 114.281 | 283.255 |
| P1,2[MW] | 50 | 50.0000 | 52.263 | 108.768 | 145.408 | 156.508 |
| P1,3[MW] | 50 | 50.0000 | 66.46 | 80.7334 | 91.9783 | 115.813 |
| P2,1[MW] | 80 | 101.9249 | 80.0 | 153.983 | 185.284 | 261.211 |
| P2,2[MW] | 50 | 50.0000 | 121.84 | 82.241 | 95.1919 | 72.6066 |
| P2,3[MW] | 50 | 57.5588 | 58.094 | 81.4515 | 77.3408 | 83.089 |
| Pw[MW] | 290.5184 | 290.5184 | 290.52 | 290.518 | 290.518 | 290.518 |
| PL[MW] | 2.9584 | 3.3925 | 5.274 | 7.43403 | 10.2943 | 20.8807 |
| $F_C$[$/h] | 4854.9860 | 5104.9225 | 5936.3 | 6750.67 | 7569.49 | 9797.22 |
| $E_T$[Kg/h] | 223.2592 | 235.9724 | 309.92 | 322.468 | 410.726 | 923.159 |
| CEED[Kg/h] | 7285.6285 | 7639.6617 | 8909.10 | 10151.5 | 11803.5 | 16560.8 |
| Computational time(s) | | | | | | 30.14 s |

The solution of the MAWTEED problem is shown in Table 6.1 above. The real power generators (P1,1-P1,3) and (P2,1-P2,3) units represent the thermal generators for area 1 and area 2, wind (Pw) generators represent the wind power generators for all three wind parks in [MW], fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h]. The obtained fuel cost using PSO is compared with Artificial Bee Colony Optimization (ABCO), Differential Evolution (DE), Evolutionary Programming (EP), Real-coded Genetic Algorithm (RCGA), and Semi-define Programming (SDP)(Basu, 2013), and (Alli, 2024) algorithms that are found in the literature, and proposed the LMM that was found in Chapter 5, as shown in Table 6.2 below.

**Table 6.2:** **PSO and LM compared with other optimisation algorithms for the MAWTEED problem using PD = 1263[MW]**

| Reference | (Basu.,2013) | | | | (Alli,2024) | Proposed LMM | Proposed PSO |
|---|---|---|---|---|---|---|---|
| **Algorithm** | **ABCO** | **DE** | **EP** | **RCGA** | **SDP** | | |
| P1,1[MW] | 500.0000 | 500.0000 | 500.0000 | 500.0000 | 311.0838 | 232.7670 | 283.255 |
| P1,2[MW] | 200.0000 | 200.0000 | 200.0000 | 200.0000 | 200.0000 | 120.1745 | 156.508 |
| P1,3[MW] | 149.9997 | 150.0000 | 149.9919 | 149.6328 | 150.0000 | 96.7054 | 115.813 |
| P2,1[MW] | 204.3358 | 204.3341 | 206.4493 | 205.9398 | 162.8737 | 238.7650 | 261.211 |
| P2,2[MW] | 154.9954 | 154.7048 | 154.8892 | 155.8322 | 159.5473 | 169.1088 | 72.6066 |
| P2,3[MW] | 67.2915 | 67.5770 | 65.2717 | 65.2209 | 101.7656 | 110.7344 | 83.089 |
| Pw [MW] | - | - | - | - | - | 313.9185 | 290.518 |
| $P_L$[MW] | 13.6224 | 13.6159 | 13.6021 | 13.6257 | - | 19.1629 | **20.8807** |
| $F_C$[$/h] | 12255.39 | 12255.42 | 12255.43 | 12256.23 | 10722.00 | 9819.9287 | **9797.22** |
| %Deviation of Fc for LMM in relation to the literature | 5.57 | 5.57 | 5.57 | 5.57 | 2.25 | 0.057 | 4.10 Average |
| $E_T$[Kg/h] | - | - | - | - | - | 830.2617 | 923.159 |
| CEED[Kg/h] | - | - | - | - | - | 16427.6875 | 16560.8 |

Figure 6.2 illustrates the relationship between fuel cost [$/h] and power demand [MW] for PSO and other optimisation techniques, outlined in Table 6.2 above.

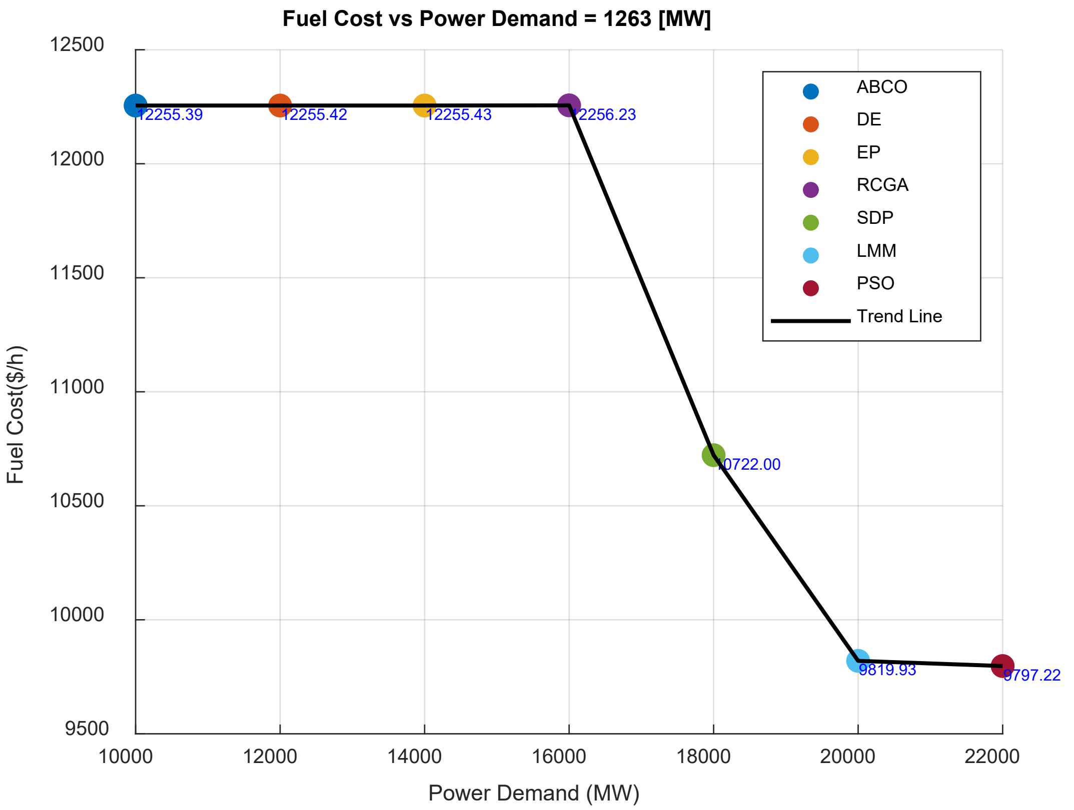



**Figure 6.2:** **Fuel cost versus power demand for different optimisation methods using the PSO algorithm for a 6-unit system**

#### 6.4.1.1 The MAWTEED problem results, and discussion for the 6-Unit System

This system utilises three wind farm networks, which supply 330 [MW] to the power dispatch problem. The following parameters for the PSO algorithm are used. Swarm Size is 80, maximum number of iterations is 600, initial weight w is 0.72, and c1, c2 are both 1.49. The proposed algorithm for a 6-unit multi-area system is compared with the LM method, and other optimization algorithms in the literature, namely, Artificial Bee Colony Optimization (ABCO), Differential Evolution (DE), Evolutionary Programming (EP), Real-coded Genetic Algorithm (RCGA), and Semi-define Programming (SDP)(Basu, 2013), and (Alli, 2024). The comparison is of the simulation results shown in Table 5.2 above.

The fuel cost results for the ABCO, DE, EP, RCGA, SDP, and LM algorithms are 5.57%, 5.57%, 5.57%, 5.57%, 2.25%, and 0.057%, respectively, compared to the PSO algorithm. When using the PSO algorithm, there is a 1–4% reduction in fuel consumption when wind power is added to the economic emission dispatch, compared with other optimisation algorithms.

### 6.4.2 Test System 2: IEEE30 bus system with 12 generating units.

This test system consists of a standard IEEE 30-bus system with four areas, 12 thermal generators (3 in each area), and a three-wind-park network that supplies 110MW of wind power to the entire dispatch problem. The total power demand is 2090 [MW]. The area demand is considered to be 500 [MW] for area 1, 410 [MW] for area 2, 580 [MW] for area 3, and 600 [MW] for area 4.  The fuel cost and wind parameters are formed based on the ones reported in (Krishnamurthy & Tzoneva, 2016) and (Alli, 2024) As a test unit, several power demand values are presented in Table 6.3 to test the system's behaviour to changes in power demand. The MATLAB script is called MAWTEED_casePSO12units. m, and the software is provided in Appendix D2.

**Table 6.3:** **The MAWTEED problem using PSO with various power demands for a 12-unit system.**

| PD[MW] | 1500 | 1600 | 1800 | 1900 | 2090 |
|---|---|---|---|---|---|
| P1,1[MW] | 35.0000 | 35.0000 | 35.0000 | 55.3949 | 58.5930 |
| P1,2[MW] | 130.0000 | 130.0000 | 130.0000 | 130.0000 | 175.3381 |
| P1,3[MW] | 125.0000 | 125.0000 | 125.0000 | 125.0000 | 138.4997 |
| P2,1[MW] | 10.0000 | 10.0000 | 10.0000 | 10.0000 | 53.1994 |
| P2,2[MW] | 49.4221 | 69.3285 | 92.1409 | 100.7025 | 77.3587 |
| P2,3[MW] | 125.0000 | 125.0000 | 125.2665 | 132.6416 | 165.1398 |
| P3,1[MW] | 108.2540 | 146.1657 | 148.5243 | 140.9480 | 152.7173 |
| P3,2[MW] | 78.1069 | 80.8725 | 161.7085 | 101.5956 | 128.8915 |
| P3,3[MW] | 50.0000 | 50.0000 | 50.0000 | 146.3462 | 182.5738 |
| P4,1[MW] | 82.4448 | 110.7047 | 118.6657 | 146.6789 | 175.0000 |
| P4,2[MW] | 146.6411 | 160.4819 | 102.8930 | 119.1091 | 189.9407 |
| P4,3[MW] | 50.0000 | 50.0000 | 185.3564 | 171.3252 | 153.4674 |
| Pw[MW] | 330.3487 | 330.3487 | 330.3487 | 330.3487 | 330.3487 |
| PL[MW] | 11.4070 | 13.7845 | 19.2397 | 21.4340 | 29.7363 |
| $F_C$[$/h] | 61402.9487 | 68225.1214 | 78007.8138 | 84856.2654 | 102939.913 |
| $E_T$[Kg/h] | 544.3535 | 657.5582 | 754.4482 | 818.4158 | 1103.8629 |
| CEED[Kg/h] | 97366.9480 | 111456.1810 | 126874.800 | 137412.0394 | 176832.189 |

The solution of the MAWTEED problem is shown in Table 6.3 above. The real power generators (P1,1-P1,3), (P2,1-P2,3), (P3,1-P3,3), and (P4,1-P4,3) units represent the thermal generators for area 1, area 2, area 3, and area 4. Wind (Pw) generators represent the wind power generators for all three wind parks in [MW], fuel cost ($F_C$), total emission ($E_T$) in [ kg/h], and CEED in [Kg/h]. The obtained fuel cost and emission using PSO is compared with Lagrange's Decomposition-Coordinating Method (LDCM), Particle Swarm Optimisation (PSO), and Simplex-Particle Swarm Optimisation (SPSO)(Chopra et al., 2018) algorithms found in the literature, as shown in Table 6.4 below.

**Table 6.4:PSO compared with other optimisation algorithms for the MAWTEED problem,**
**PD = 2090[MW]**

| Algorithm | LDCM | PSO | SPSO | LMM (Krishnamurthy,2011) | Proposed LMM | Proposed PSO |
|---|---|---|---|---|---|---|
| | (Chopra et al., 2018) | | | | | |
| P1,1[MW] | 131.45 | 160.9 | 165 | 182.7879 | 87.5974 | 58.5930 |
| P1,2[MW] | 209.49 | 163.3 | 260 | 302.6497 | 138.5237 | 175.3381 |
| P1,3[MW] | 202.25 | 289.4 | 265 | 287.4964 | 133.0488 | 138.4997 |
| P2,1[MW] | 150 | 109.6 | 150 | 130.5891 | 86.1913 | 53.1994 |
| P2,2[MW] | 110 | 117.1 | 75 | 110.0000 | 110.0000 | 77.3587 |
| P2,3[MW] | 191.63 | 184.4 | 175 | 147.6606 | 125.0000 | 165.1398 |
| P3,1[MW] | 175 | 171.5 | 175 | 142.9309 | 146.1331 | 152.7173 |
| P3,2[MW] | 215 | 197.5 | 160 | 181.2219 | 182.9771 | 128.8915 |
| P3,3[MW] | 236.38 | 198.3 | 280 | 168.9267 | 173.4331 | 182.5738 |
| P4,1[MW] | 164.09 | 230 | 175 | 126.3394 | 141.6098 | 175.0000 |
| P4,2[MW] | 180.2 | 144.8 | 160 | 140.9621 | 156.0605 | 189.9407 |
| P4,3[MW] | 306.18 | 211.5 | 280 | 236.9631 | 262.9685 | 153.4674 |
| Pw[MW] | - | - | - | - | 329.2499 | 330.3487 |
| Fc[$] | 144058.2 | 136598.3 | 134557.6 | 125883.5243 | 106977.016 | 102939.913 |
| %Deviation of Fc for LMM in relation to the literature | 8.3236 | 7.02568 | 6.65642 | 5.01339 | 0.9616 | 5.59614 |

| | | | | | | |
|---|---|---|---|---|---|---|
| PL[MW] | 61.82 | 48.22 | 43.804 | 68.5286 | 33.6248 | 29.7363 |
| ET[T] | 1923.7 | 3713.93 | 1473.935 | 1684.6603 | 1120.2618 | 1103.8629 |
| %Deviation of $E_T$ for LMM in relation to the literature | 13.5396 | 27.0878 | 7.17807 | 10.4141 | 0.36866 | 11.7176 |
| CEED[$/h] | 266400.92 | 277258.2 | 232500.67 | 225665.1721 | 181987.099 | 176832.189 |
| %Deviation of CEED for LMM in relation to the literature | 10.104 | 11.0579 | 6.7999 | 6.06625 | 0.71832 | 6.94927 |

Figures 6.3, 6.4, and 6.5 graphically illustrate the relationship between fuel cost [$/h], emission [Kg/h], and CEED with power demand [MW] for various optimisation strategies.

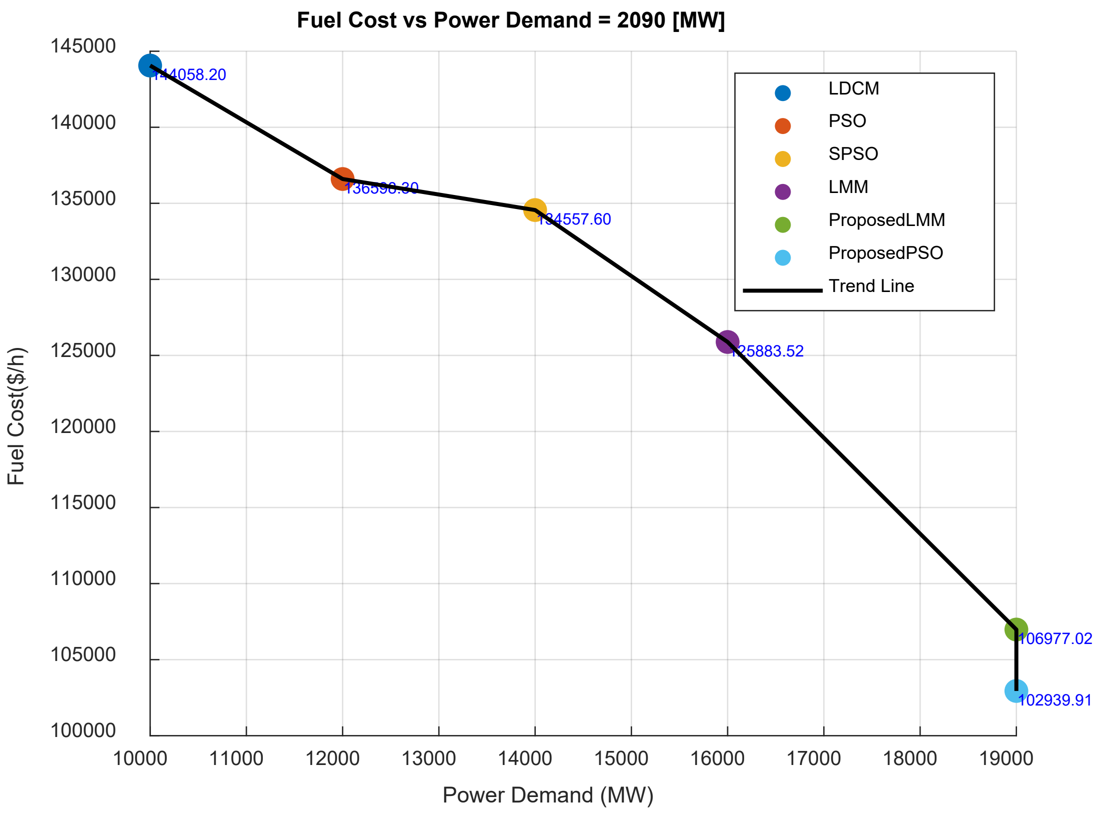



**Figure 6.3**: **Values of fuel cost and power demand for various optimisation strategies applied to the PSO algorithm for a 12-unit system**

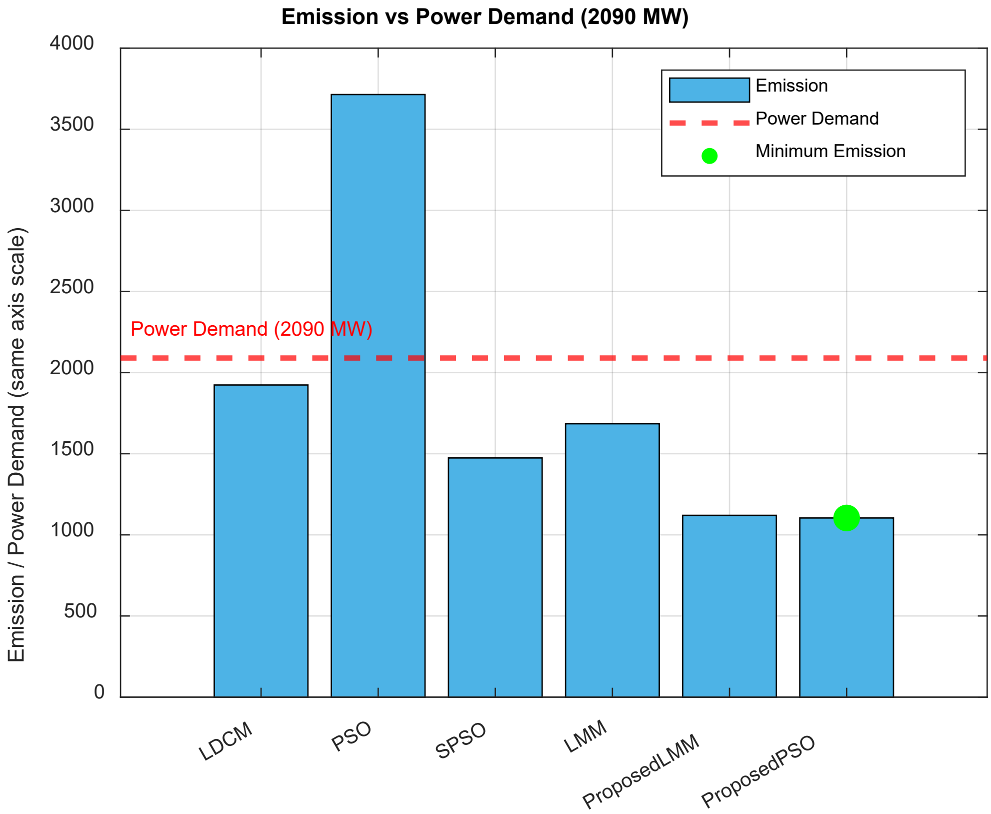



**Figure 6.4:** **Emission versus power demand for different optimisation methods using the PSO algorithm for a 12-unit system**

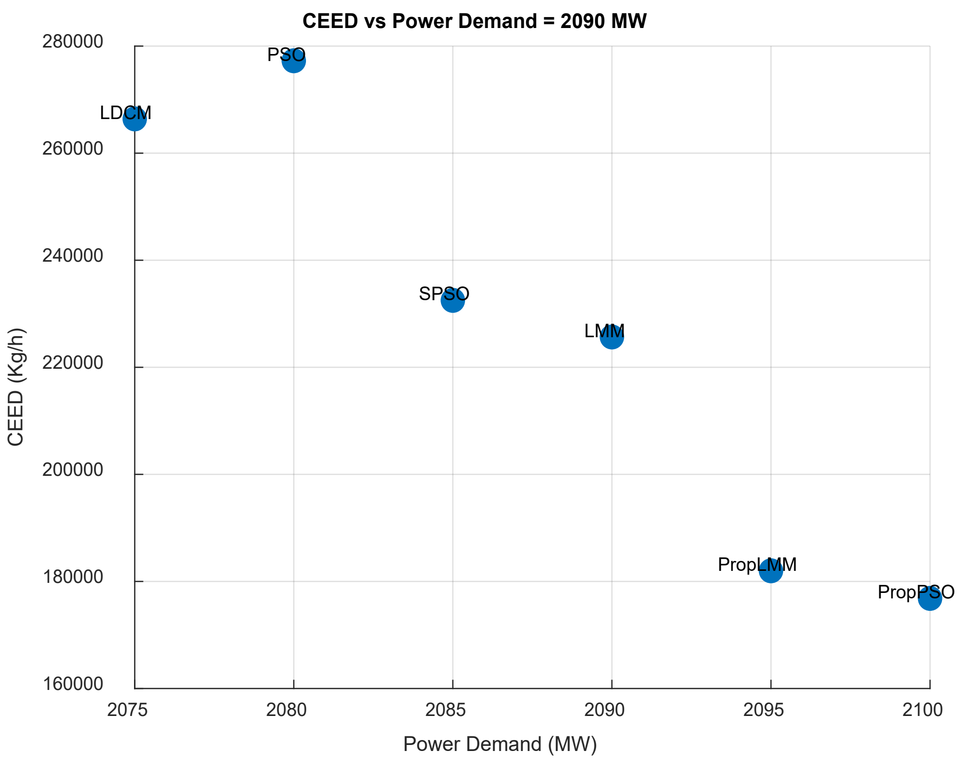



**Figure 6.5:** **CEED versus power demand for different optimisation methods using the PSO algorithm for a 12-unit system**

#### 6.4.2.1 The MAWTEED problem results, and discussion for 12-Unit System

The PSO is applied for solving the multi-area economic emission dispatch problem with wind power, utilizing three wind farms of 110MW each. The following parameters for the PSO algorithm are used. Swarm Size is 100, maximum number of iterations is 500, initial weight w is 0.9, and c1, c2 are both 2. The PSO is compared with Lagrange's Decomposition-Coordinating Method (LDCM), Particle Swarm Optimisation (PSO), and Simplex-Particle Swarm Optimisation (SPSO)(Chopra et al., 2018)and LMM developed in Chapter 5. The fuel cost results for LDCM, PSO, SPSO, LMM, and the developed LMM algorithms are 8.32%, 7.026%, 6.67%, 5.01%, and 0.96%, respectively, compared to the proposed PSO algorithm. When using the PSO algorithm, there is a 1–6% reduction in fuel consumption when wind power is added to the economic emission dispatch, compared with other optimisation algorithms. The resultant NOx emissions are 13.54%, 27.09%, 7.18%, 10.41%, and 0.37% lower than those of the developed PSO algorithm. The PSO algorithm has effectively reduced pollutant levels by 1–12% compared to other optimization algorithms.

The proposed algorithm has lower fuel cost and lower emissions and is observed to be superior to all other optimization algorithms.

### 6.4.3 Test System 3: IEEE30 bus system with 40 generating units.

The problem of solving MAWTEED using the Lagrange algorithm is carried out on a standard IEEE 30-bus two-area system with 40 thermal generators (20 in each area) and a three-wind park network supplying 110MW each to the entire dispatch problem. The total power demand is considered as 10500MW. The area demand is considered as 7500 MW for area 1 and 3000 MW for area 2, respectively. The fuel cost and wind parameters are formed based on the ones reported in (Krishnamurthy & Tzoneva, 2016) and (Alli, 2024) and are shown in Tables 6.5 and 6.6.

**Table 6.5:** Wind-speed data

| $v_{in}$ Cut-in wind speed | $v_r$ Rated wind-speed | $v_o$ Cut-out wind speed | $w_r$ Rated Power | $C_{pw}$ Underestimation | $C_{rw}$ Overestimation | $d_w$ direct cost |
|---|---|---|---|---|---|---|
| 5 | 15 | 45. | 330 | 5 | 5 | 0 |

**Table 6.6:** **IEEE 30 Bus System data for 40 thermal systems.**

| Generating Unit [MW] | Generator Limits | | Fuel cost coefficients | | | Emission coefficients | | |
|---|---|---|---|---|---|---|---|---|
| | $P_{min}$ | $P_{max}$ | $a_i$ [$/(MWh)$^2$] | $b_i$ [$/MWh] | $c_i$ [$/h] | $\alpha_i$ [t/(MWh)$^2$] | $\beta_i$ [t/MWh] | $\gamma_i$ [t/h] |
| 1 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 2 | 36 | 114 | 0.0069 | 6.73 | 94.705 | 0.048 | −2.22 | 60 |
| 3 | 60 | 120 | 0.02028 | 7.07 | 309.54 | 0.0762 | −2.36 | 100 |
| 4 | 80 | 190 | 0.00942 | 8.18 | 369.03 | 0.054 | −3.14 | 120 |
| 5 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 6 | 68 | 140 | 0.01142 | 8.05 | 222.33 | 0.0854 | −3.08 | 80 |
| 7 | 110 | 300 | 0.00357 | 8.03 | 287.71 | 0.0242 | −3.06 | 100 |
| 8 | 135 | 300 | 0.00492 | 6.99 | 391.98 | 0.031 | −2.32 | 130 |
| 9 | 135 | 300 | 0.00573 | 6.6 | 455.76 | 0.0335 | −2.11 | 150 |
| 10 | 130 | 300 | 0.00605 | 12.9 | 722.82 | 0.425 | −4.34 | 280 |
| 11 | 94 | 375 | 0.00515 | 12.9 | 635.2 | 0.0322 | −4.34 | 220 |
| 12 | 94 | 375 | 0.00569 | 12.8 | 654.69 | 0.0338 | −4.28 | 225 |
| 13 | 125 | 500 | 0.00421 | 12.5 | 913.4 | 0.0296 | −4.18 | 300 |
| 14 | 125 | 500 | 0.00752 | 8.84 | 1760.4 | 0.0512 | −3.34 | 520 |
| 15 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 16 | 125 | 500 | 0.00708 | 9.15 | 1728.3 | 0.0496 | −3.55 | 510 |
| 17 | 220 | 500 | 0.00313 | 7.97 | 647.85 | 0.0151 | −2.68 | 220 |
| 18 | 220 | 500 | 0.00313 | 7.95 | 649.69 | 0.0151 | −2.66 | 222 |
| 19 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |
| 20 | 242 | 550 | 0.00313 | 7.97 | 647.81 | 0.0151 | −2.68 | 220 |
| 21 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 290 |
| 22 | 254 | 550 | 0.00298 | 6.63 | 785.96 | 0.0145 | −2.22 | 285 |
| 23 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 24 | 254 | 550 | 0.00284 | 6.66 | 794.53 | 0.0138 | −2.26 | 295 |
| 25 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 26 | 254 | 550 | 0.00277 | 7.1 | 801.32 | 0.0132 | −2.42 | 310 |
| 27 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 28 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 29 | 10 | 150 | 0.52124 | 3.33 | 1055.1 | 1.842 | −1.11 | 360 |
| 30 | 47 | 97 | 0.0114 | 5.35 | 148.89 | 0.085 | −1.89 | 50 |
| 31 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 32 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 33 | 60 | 190 | 0.0016 | 6.43 | 222.92 | 0.0121 | −2.08 | 80 |
| 34 | 90 | 200 | 0.0001 | 8.95 | 107.87 | 0.0012 | −3.48 | 65 |
| 35 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 36 | 90 | 200 | 0.0001 | 8.62 | 116.58 | 0.0012 | −3.24 | 70 |
| 37 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 38 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 39 | 25 | 110 | 0.0161 | 5.88 | 307.45 | 0.095 | −1.98 | 100 |
| 40 | 242 | 550 | 0.00313 | 7.97 | 647.83 | 0.0151 | −2.68 | 220 |

Several power demand values are presented in Table 6.7 to validate the system's performance under alterations in power demand. The MATLAB script is identified as WTEED_casestudy9. m, and the software is available in Appendix D3. The solution of the MAWTEED problem is shown in Table 6.7. The real power generators (P1,1-P1,20) units represent the thermal generators for area 1, generators from (P2,1-2,20) represent generators for area 2, and wind (Pw) generators represent only wind power for area1 all in [MW], fuel cost $(F_C)$, total emission $(E_T)$ in [ kg/h], and CEED in [Kg/h].

**Table 6.7:** **The MAWTEED problem PSO method with various power demands for a 40-unit system**

| PD[MW] | 8000 | 8500 | 9000 | 9500 | 10000 | 10500 |
|---|---|---|---|---|---|---|
| P1,1[MW] | 36.0000 | 114.0000 | 114.0000 | 114 | 114.0000 | 114.0000 |
| P1,2[MW] | 36.0000 | 114.0000 | 114.0000 | 36 | 36.0000 | 114.0000 |
| P1,3[MW] | 60.0000 | 102.8100 | 115.0000 | 120 | 120.0000 | 60.0000 |
| P1,4[MW] | 190.0000 | 190.0000 | 190.0000 | 190 | 80.0000 | 190.0000 |
| P1,5[MW] | 97.0000 | 97.0000 | 47.0000 | 97 | 97.0000 | 97.0000 |
| P1,6[MW] | 134.1925 | 112.6848 | 140.0000 | 140 | 140.0000 | 140.0000 |
| P1,7[MW] | 110.0000 | 300.0000 | 300.0000 | 300 | 300.0000 | 300.0000 |
| P1,8[MW] | 135.0000 | 175.5269 | 300.0000 | 300 | 300.0000 | 300.0000 |
| P1,9[MW] | 300.0000 | 300.0000 | 300.0000 | 300 | 300.0000 | 300.0000 |
| P1,10[MW] | 300.0000 | 300.0000 | 300.0000 | 300 | 300.0000 | 300.0000 |
| P1,11[MW] | 216.6743 | 299.4848 | 375.0000 | 375 | 375.0000 | 375.0000 |
| P1,12[MW] | 375.0000 | 375.0000 | 375.0000 | 375 | 375.0000 | 375.0000 |
| P1,13[MW] | 314.6343 | 500.0000 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,14[MW] | 500.0000 | 476.6063 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,15[MW] | 500.0000 | 500.0000 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,16[MW] | 500.0000 | 342.3620 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,17[MW] | 399.2286 | 370.5245 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,18[MW] | 500.0000 | 500.0000 | 500.0000 | 500 | 500.0000 | 500.0000 |
| P1,19[MW] | 516.2649 | 550.0000 | 550.0000 | 550 | 550.0000 | 550.0000 |
| P1,20[MW] | 550.0000 | 550.0000 | 550.0000 | 550 | 550.0000 | 550.0000 |
| P2,1[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 453.9478 | 301.2412 |
| P2,2[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 254.0000 | 254.0000 |
| P2,3[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 254.0000 | 254.0000 |
| P2,4[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 254.0000 | 254.0000 |
| P2,5[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 254.0000 | 254.0000 |
| P2,6[MW] | 254.0000 | 254.0000 | 254.0000 | 254 | 254.0000 | 254.0000 |
| P2,7[MW] | 10.0000 | 10.0000 | 10.0000 | 10 | 10.0000 | 10.0000 |
| P2,8[MW] | 10.0000 | 10.0000 | 10.0000 | 10 | 10.0000 | 10.0000 |
| P2,9[MW] | 10.0000 | 10.0000 | 10.0000 | 10 | 41.9918 | 10.0000 |
| P2,10[MW] | 47.0000 | 47.0000 | 47.0000 | 47 | 97.0000 | 97.0000 |
| P2,11[MW] | 60.0000 | 60.0000 | 60.0000 | 60 | 116.3246 | 190.0000 |
| P2,12[MW] | 60.0000 | 60.0000 | 60.0000 | 190 | 184.7120 | 189.4034 |
| P2,13[MW] | 60.0000 | 60.0000 | 60.0000 | 60 | 75.4467 | 125.3571 |
| P2,14[MW] | 90.0000 | 90.0000 | 90.0000 | 90 | 90.0000 | 90.0000 |
| P2,15[MW] | 90.0000 | 90.0000 | 90.0000 | 90 | 90.0000 | 200.0000 |
| P2,16[MW] | 90.0000 | 90.0000 | 90.0000 | 90 | 143.5606 | 90.0000 |
| P2,17[MW] | 25.0000 | 25.0000 | 25.0000 | 25 | 25.0000 | 25.0000 |
| P2,18[MW] | 25.0000 | 25.0000 | 25.0000 | 25 | 25.0000 | 25.0000 |
| P2,19[MW] | 25.0000 | 25.0000 | 25.0000 | 25 | 25.0000 | 25.0000 |
| P2,20[MW] | 242.0000 | 242.0000 | 242.0000 | 242 | 242.0000 | 242.0000 |
| Pw[MW] | 330 | 330 | 330 | 330 | 330 | 330 |
| $F_C$[$/h] | 106104.2224 | 111354.5625 | 117996.8615 | 118687.8507 | 121485.3538 | 121665.9732 |
| $P_L$[MW] | - | | | | | |
| $E_T$[Kg/h] | 99759.4099 | 99968.6527 | 111461.7101 | 111791.3554 | 115512.0981 | 112105.2725 |
| CEED[Kg/h] | 184679.0933 | 197631.2948 | 217073.7455 | 219149.4535 | 224408.7740 | 225669.8748 |

The wind-thermal multi-area economic emission dispatch problem using the PSO algorithm in relation to other optimization algorithms, the Salp Swarm Algorithm(SSA), the Flower Pollination Algorithm(FPA), and the Backtracking Search Algorithm(BSA) (Alli, 2024), and Hybridizing Sum-Local Search Optimize (HSLSO)(Chaudhary et al., 2020) and are given in Table 6.8.

**Table 6.8:** **MAWTEED using PSO in relation to other optimization algorithms applied 40-unit system, PD = 10500[MW]**

| **Algorithm** | **SSA** | **FPA** | **BSA** | **HSLSO** | **Developed** LMM **330[MW] wind power** | **Developed PSO 330MW wind power** |
|---|---|---|---|---|---|---|
| **Reference** | **(Alli.,2024)** | | | **(Chaudhary.,2020)** | | |
| P1,1[MW] | 113.9998 | 97.4000 | 97.3999 | 110.8012 | 114.0000 | 114.0000 |
| P1,2[MW] | 113.9996 | 179.7331 | 179.7331 | 113.9997 | 114.0000 | 114.0000 |
| P1,3[MW] | 120 | 87.7999 | 87.7999 | 120.0 | 120.0000 | 60.0000 |
| P1,4[MW] | 179.73331 | 105.3999 | 105.3999 | 179.7331 | 184.4384 | 190.0000 |
| P1,5[MW] | 96.0324 | 300 | 259.5996 | 95.551 | 97.0000 | 97.0000 |
| P1,6[MW] | 140 | 284.5997 | 284.5996 | 140.0 | 140.0000 | 140.0000 |
| P1,7[MW] | 300 | 295.3663 | 284.5997 | 300.0 | 300.0000 | 300.0000 |
| P1,8[MW] | 284.5995 | 279.5997 | 204.7997 | 284.5997 | 300.0000 | 300.0000 |
| P1,9[MW] | 284.6002 | 94 | 94 | 284.5997 | 300.0000 | 300.0000 |
| P1,10[MW] | 269.9999 | 94 | 94 | 270.0 | 194.2785 | 300.0000 |
| P1,11[MW] | 168.7999 | 214.7598 | 304.5195 | 94.0 | 239.9567 | 375.0000 |
| P1,12[MW] | 350.0002 | 394.2794 | 394.2793 | 300.0 | 238.2515 | 375.0000 |
| P1,13[MW] | 394.2794 | 394.2794 | 394.2793 | 304.5195 | 326.5568 | 500.0000 |
| P1,14[MW] | 394.2793 | 394.2794 | 394.2793 | 394.2797 | 359.8966 | 500.0000 |
| P1,15[MW] | 304.5197 | 489.2794 | 489.2794 | 484.0395 | 358.8979 | 500.0000 |
| P1,16[MW] | 484.0391 | 399.5196 | 489.2793 | 484.0391 | 358.8979 | 500.0000 |
| P1,17[MW] | 489.2794 | 511.2794 | 511.2794 | 489.2794 | 500.0000 | 500.0000 |
| P1,18[MW] | 489.2796 | 511.2794 | 511.2793 | 489.2796 | 500.0000 | 500.0000 |
| P1,19[MW] | 511.2794 | 511.2794 | 523.2794 | 549.9998 | 550.0000 | 550.0000 |
| P1,20[MW] | 511.2793 | 511.2794 | 523.2794 | 511.2791 | 550.0000 | 550.0000 |
| P2,1[MW] | 523.2795 | 523.2794 | 523.2794 | 523.2792 | 477.1028 | 301.2412 |
| P2,2[MW] | 343.7598 | 523.2794 | 523.2793 | 523.2791 | 476.5912 | 254.0000 |
| P2,3[MW] | 254 | 523.2794 | 523.2794 | 523.2794 | 478.2959 | 254.0000 |
| P2,4[MW] | 523.2794 | 523.2794 | 523.2794 | 523.2794 | 478.2959 | 254.0000 |
| P2,5[MW] | 523.2793 | 10 | 10 | 523.2795 | 453.0060 | 254.0000 |
| P2,6[MW] | 523.2793 | 10 | 10 | 254.0 | 453.0060 | 254.0000 |
| P2,7[MW] | 109.9999 | 10 | 10 | 10.0001 | 11.0158 | 10.0000 |
| P2,8[MW] | 109.9999 | 87.7999 | 87.7999 | 10 | 11.0158 | 10.0000 |
| P2,9[MW] | 110 | 190 | 190 | 10 | 11.0158 | 10.0000 |
| P2,10[MW] | 87.7998 | 190 | 190 | 87.7997 | 97.0000 | 97.0000 |
| P2,11[MW] | 159.733 | 190 | 190 | 188.5959 | 477.1028 | 190.0000 |
| P2,12[MW] | 159.733 | 200 | 196.9619 | 159.7331 | 476.5912 | 189.4034 |
| P2,13[MW] | 159.7331 | 164.7999 | 164.7998 | 159.733 | 478.2959 | 125.3571 |
| P2,14[MW] | 90 | 164.7999 | 164.7998 | 164.8002 | 478.2959 | 90.0000 |
| P2,15[MW] | 164.8 | 110 | 109.9997 | 164.7998 | 453.0060 | 200.0000 |
| P2,16[MW] | 164.8 | 110 | 89.1141 | 164.7998 | 453.0060 | 90.0000 |
| P2,17[MW] | 72.296 | 110 | 109.9997 | 89.1143 | 11.0158 | 25.0000 |
| P2,18[MW] | 89.114 | 511.2794 | 511.2793 | 89.114 | 11.0158 | 25.0000 |
| P2,19[MW] | 89.114 | 550 | 549.9992 | 89.1134 | 11.0158 | 25.0000 |
| P2,20[MW] | 242 | 550 | 550 | 242.0001 | 97.0000 | 242.0000 |
| Pw[MW] | - | - | - | - | 282.5545 | 330 |
| Fc[$/h] | 120857.2447 | 130573.2847 | 130580.7421 | 125100.2621 | 120069.1486 | 121665.9732 |
| %Deviation of Fc for LMM in relation to the literature | 0.166732 | 1.76565 | 1.76707 | 0.69586 | 0.330284 | 0.945119 |
| PL[MW] | | | | | - | |
| ET[Kg/h] | | | | | 79614.2253 | 112105.2725 |
| CEED[Kg/h] | - | - | - | - | 214795.2613 | 225669.8748 |

Figure 6.6 shows the relationship between the fuel cost [$/h] and the power demand [MW] for several optimisation methods.

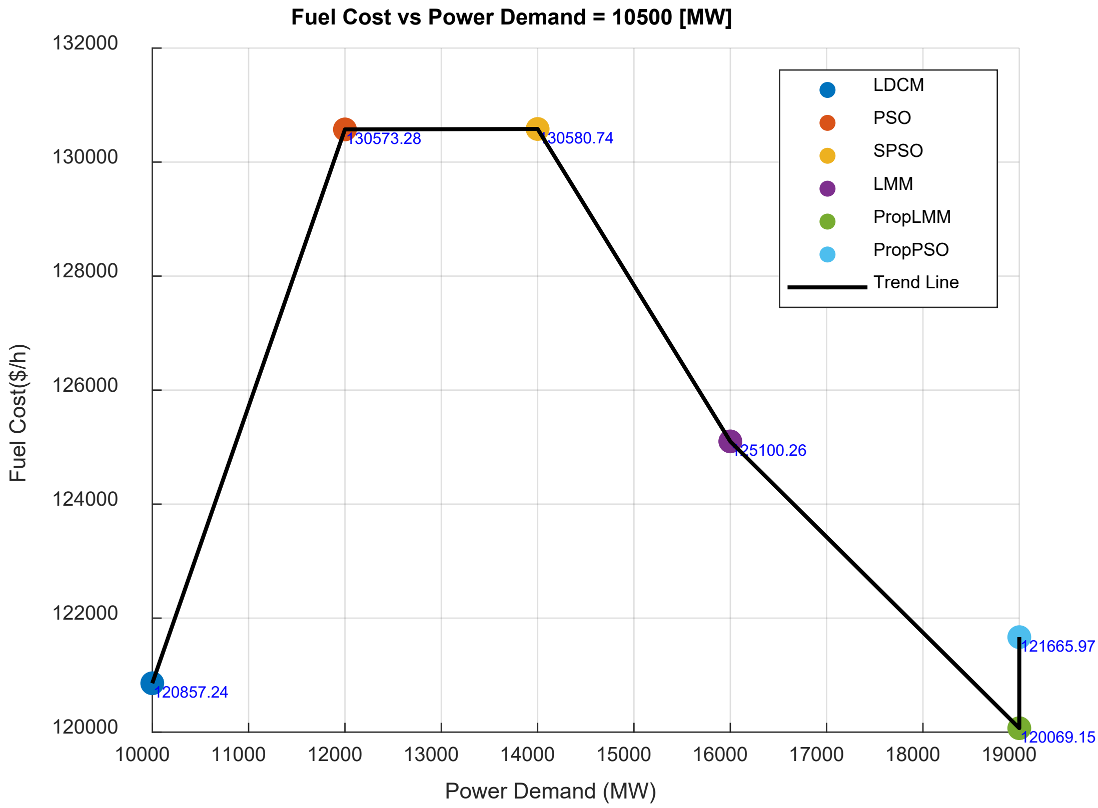



**Figure 6.6:** **Values of fuel cost and power demand for the MAWTEED problem applied to PSO with a 40-unit system**

As shown in Table 6.9, the fuel cost and emission figures are further compared with the statistical results for various algorithms.

**Table 6.9:** **Relation of PSO with various algorithms (Ahmed et al., 2022)**

| Algorithm | Best cost ($/h) | Best cost (%) Deviation | Best Emission (Kg/h) | Best Emission (%) Deviation |
|---|---|---|---|---|
| ABC | 126 480.56 | 0.97011 | 209285.74 | 15.1187 |
| EMA | 125910.69 | 0.85725 | 210238.19 | 15.2218 |
| NSGA-II | 125830.00 | 0.84123 | 210950.00 | 15.2984 |
| SSA-WSA | 125760.05 | 0.82733 | 206705.97 | 14.8365 |
| MOSSA | 125591.29 | 0.79377 | 205965.40 | 14.7546 |
| CSOA | 124330.50 | 0.54158 | 116560.5 | 0.97418 |
| **Developed LMM** | **120069.1486** | **0.330284** | **79614.2253** | **8.47359** |
| **Developed PSO** | **121665.9732** | **0.80521** | **112105.2725** | **12.7007** |

The best fuel [$/h] and emission [Kg/h] values for different optimization algorithms are compared with those of the PSO algorithm, as shown in Table 6.11 above. The best emission values are plotted against power demand [MW] as illustrated in Figure 6.7 below.

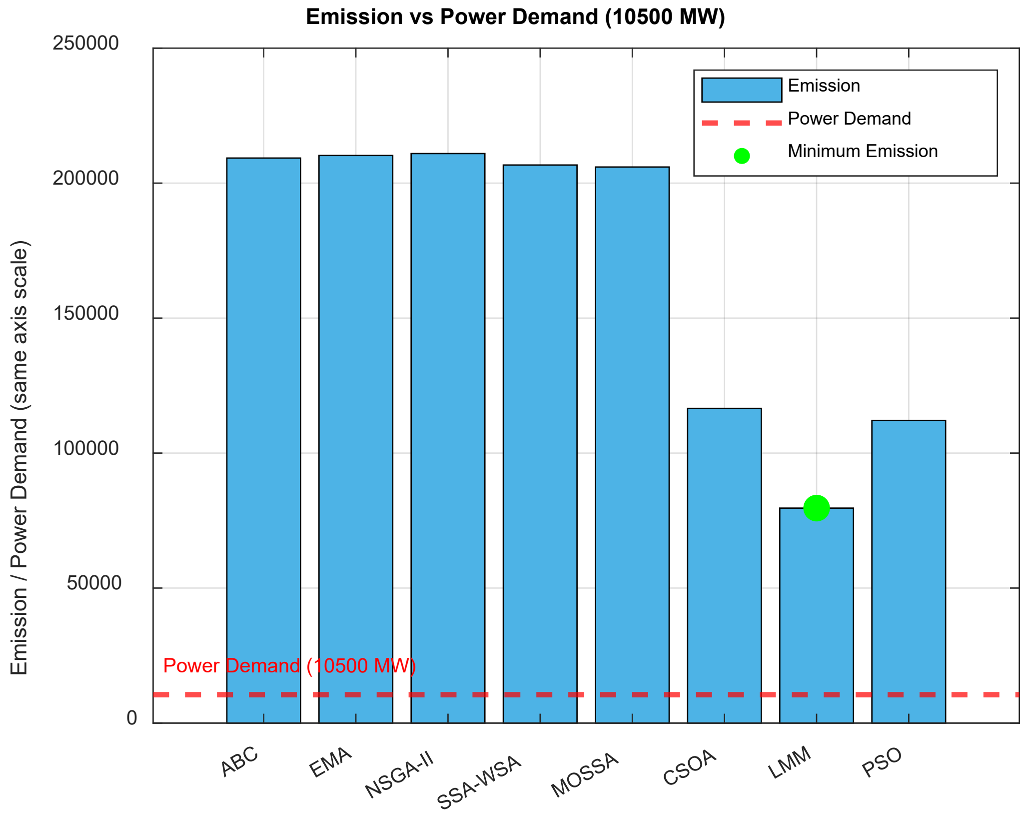



**Figure 6.7:** **Emission versus power demand for the bi-criteria dispatch problem using PSO for a 40-unit system**

#### 6.4.3.1 The MAWTEED problem results and discussion for the 40-Unit System

The studied system includes a fourth generating unit to illustrate the effectiveness of the PSO algorithm compared to other optimization methods in minimizing fuel costs and emissions. The parameters used for the PSO algorithm are as follows. Swarm Size is 100, maximum number of iterations is 1000, initial weight w is 0.9, and c1, c2 are both 2.  The PSO algorithm is compared with different heuristic algorithms, i.e., Salp Swarm Algorithm (SSA), Flower Pollination Algorithm (FPA), and Backtracking Search Algorithm (BSA) (Alli, 2024), and Hybridizing Sum-Local Search Optimize (HSLSO)(Chaudhary et al., 2020). The comparison is made based on the best cost and lowest emissions. The summary of the comparison between the PSO algorithm and the mentioned heuristic algorithms is given in Table 6.11.

Table 6.11 shows that the PSO values for fuel cost and emission minimization are 1% and 13% better, respectively, than those for the MAWTEED problem.

## 6.5 Conclusions

In this chapter, a Multi-Area Wind-Thermal Economic Emission Dispatch problem is solved using the PSO. Fuel and emission costs are minimised in the bi-objective problem of economic emission dispatch optimisation. This chapter presents the transmission line losses, emissions, and inequality limits. Three test systems: a 6-unit system with two areas, a 12-unit system with four areas, and a 40-unit system with two areas are used to evaluate the algorithm. Compared to other methods, the fuel cost for 6-unit systems is 1-5% cheaper. Additionally, compared with previous optimisation techniques, the 12-unit system's fuel cost and emissions are 1-6% and 1-12%, respectively. Lastly, fuel consumption and emissions for the 40-unit system have been decreased by 1% and 1-14%, respectively.

By carefully selecting the PSO parameters w, c1, c2, population size, and number of iterations, the procedure efficiently converges to an optimal solution. The fuel and emissions for the MAWTEED problem are affected by the random selection of PSO variables, and the resulting LMM and PSO values differ by 1% across all test systems. The optimal requirements for the 6-unit, 12-unit, and 40-unit systems, however, remain within the anticipated ranges for fuel cost, emissions, and total generation in the MAWTEED problem when compared to the literature. Compared to the heuristic methods described in the literature, the classical Lagrange approach utilising decomposition coordination can easily produce optimal results for fuel and emission reduction. For simple convex instances, Lagrange's traditional techniques are practical and precise.

The PSO algorithm can solve the MAWTEED problem better than the Lagrange method when the objective function and constraints are nonlinear and nonconvex.

# CHAPTER SEVEN
## CONCLUSION AND FUTURE RECOMMENDATIONS

### 7.1 Introduction

This thesis explores wind energy in the current grid to reduce emissions and generation costs. Relative works are as follows: relative contributions to the findings of chapter 2 of how literature's value of the problem differs from the results of the WTEED single-area (chapter 3) and multi-area (subsequent chapters 4) problems solved by efforts from this thesis. The WTEED single-area problem is solved in Chapter 3 using Lagrange multipliers, and in Chapter 4 using PSO, both of which are compared with the literature. The WTEED multi-area problem is solved in Chapter 5 using Lagrange multipliers, and in Chapter 6 using PSO, both of which are compared with the literature. This conclusion provides an overview synthesis of such efforts for both implementing wind energy in the current grid for reduced emissions and costs and is broken down as follows: Section 7.1, Aims and Objective, Section 7.2, Study Contribution, Section 7.3, Findings and Application, Section 7.4, Suggestions for Future Studies, and Section 7.5, Author's Note for paper publication for clarity.

### 7.2 Research aim and objectives

#### 7.2.1 Research Aim

To enhance the wind-thermal economic emission dispatch problem with formulations and algorithms, and solve it for single-area and multi-area systems. To provide a decomposition-coordinating solution to the dispatch problem via Lagrange's method. Software operates sequentially on one computer.

#### 7.2.2 Objectives

- To review the literature of already existing formulations, methods, and algorithms of wind-thermal, wind-diesel, wind-PV, and hydrothermal economic dispatch problems.
- To formulate the wind-thermal economic emission dispatch problem with constraints for the single-area power system.
- To formulate the wind-thermal economic dispatch problem with constraints for the multi-area power system.
- To formulate LaGrange's method for the wind-thermal economic dispatch problem for single and multi-area power systems.
- To formulate the PSO method for the wind-thermal economic dispatch problem for single and multi-area power systems.
- To construct MATLAB software for LaGrange's and PSO methods for the wind-thermal economic dispatch problem for the single area power systems.

- To construct MATLAB software for LaGrange's and PSO methods for the wind-thermal economic dispatch problem for the multi-area power systems based on decomposition.
- To test the constructed software on standard IEEE benchmark systems and compare the obtained solutions with existing solutions in the literature.

## 7.3 The deliverables of the thesis

### 7.3.1 A comparative review of the current methodologies for addressing the single-area and multi-area economic emission dispatch problem.

In Chapter Two, Various optimizations and algorithms for single-area and multi-area wind-thermal economic emission dispatch solutions are assessed. This study compares classical, heuristic, and hybrid optimization strategies and algorithms for economic emission dispatch problems in single- and multi-area systems. First, economic emission dispatch solutions for single and multi-area systems are comprehensively investigated and thereafter tabulated, starting with single-area WTEED, then hybrid, and finally multi-area economic emission dispatch solutions. For example, the studies are relative to the number of studies reported per year, the methods and algorithms used, and the literature review articles assessing single-area and multi-area emission dispatch with single-criterion and multi-criterion optimization for economic dispatch. The table assessing these articles goes even further, covering: System (wind, storage, coal, diesel, etc.); Objective function (single- or multi-objective); Constraints (equality and inequality); Algorithm; Software; Power systems used; and Real-world application.

### 7.3.2 Mathematical formulation of the economic dispatch problems

A formulation of the quadratic cost function and the Weibull Distribution Function (WDF) for single- and multi-area Wind-Thermal Economic Emission (WTEED) is explored to solve the WTEED for multi-criteria dispatch. Formulations for single- and multi-area dispatch are discussed in Chapters 3 and 5, respectively.

### 7.3.3 Optimisation methods developed to solve the economic emission dispatch problems

The Lagrange method for the single-area WTEED is derived in Chapter 3. The solution for multi-area WTEED is derived using the Lagrange Decomposition method from Chapter 5. AI optimization of Particle Swarm Optimization (PSO) to solve single-area WTEED is employed in Chapter 4, while PSO to solve multi-area WTEED is employed in Chapter 6. A comparison is made between the solutions derived in Chapters 3-6 and other methods in the literature applied to single- and multi-area dispatch problems..

### 7.3.4 Software development and implementation for the solution to the economic emission dispatch problem for both single and multi-area systems.

Software developed for practical implementation by engineers is based on Lagrange methods for single-area and multi-area decomposition-coordination in the Wind-Thermal Economic Emission Dispatch (WTEED). Software developed from PSO methods of solution for use by practical implementation engineers for single-area and multi-area WTEED problems. This software was designed as a solution to dispatch issues and was verified against standard IEEE benchmarks and compared with other solutions in the literature. The software for dispatching problems in single- and multi-area systems is presented in Tables 7.1 and 7.2.

**Table 7.1:** **Developed programs for the solution of a single area WTEED problem**

| Type of functions used in the WTEED problem | Algorithm | Type of Benchmark/Network | Appendix Identification | MATLAB Identification |
|---|---|---|---|---|
| Quadratic cost, emission, and Weibull Distribution Functions | Lagrange | IEEE30 Bus 6-unit system | Appendix A1 | WTEED_case6units.m |
| | | IEEE30 Bus 10-unit system | Appendix A2 | WTEED_case10units.m |
| | | IEEE30 Bus 40-unit system | Appendix A3 | WTEED_case40units.m |
| Quadratic cost, emission, and Weibull Distribution Functions | PSO | IEEE30 Bus 6-unit system | Appendix B1 | WTEED_casePSO6units.m |
| | | IEEE30 Bus 10-unit system | Appendix B2 | WTEED_casePSO10units.m |
| | | IEEE30 Bus 40-unit system | Appendix B3 | WTEED_casePSO40units.m |

The developed programs for a single-area WTEED problem are illustrated in Table 7.1.

**Table 7.2:** **Developed programs for the solution of a multi-area WTEED problem**

| Type of functions used in the WTEED problem | Algorithm | Type of Benchmark/Network | Tie line | Appendix Identity | MATLAB Identification |
|---|---|---|---|---|---|
| Quadratic cost, emission, and Weibull Distribution Functions | Lagrange | IEEE30 Bus 6-unit system<br>(i) Two areas<br>(ii) 3 Thermal units in each<br>(iii) Wind power of 330MW | 1 | Appendix C1 | MAWTEED_case6units.m |
| | | IEEE30 Bus 12-unit system<br>(i) Four areas<br>(ii) 3 Thermal units in each<br>(iii) Wind power of 330MW | 6 | Appendix C2 | MAWTEED_case10units.m |
| | | IEEE30 Bus 40-unit system<br>(i) Two areas<br>(ii) 20 Thermal units in each<br>(iii) Wind power of 330MW | 1 | Appendix C3 | MAWTEED_case40units.m |
| Quadratic cost, emission, and Weibull Distribution Functions | PSO | IEEE30 Bus 6-unit system<br>(i) Two areas<br>(ii) 3 Thermal units in each<br>(iii) Wind power of 330MW | 1 | Appendix D1 | MAWTEED_casePSO6units.m |
| | | IEEE30 Bus 12-unit system<br>(i) Four areas<br>(ii) 3 Thermal units in each<br>(iii) Wind power of 330MW | 6 | Appendix D2 | MAWTEED_casePSO10units.m |
| | | IEEE30 Bus 40-unit system<br>(i) Two areas<br>(ii) 20 Thermal units in each<br>(iii) Wind power of 330MW | 1 | Appendix D3 | MAWTEED_casePSO40units.m |

The established programs for a multi-area WTEED problem are illustrated in Table 7.2 above.

## 7.4 Implementation of the thesis findings

Some industrial facilities with comparable features can use the created techniques, algorithms, and software applications with little adjustments, such as:

- Solve the Economic dispatch problem in both single and multiple area power systems.
- Enhancement of Power Systems Economic Dispatch Based on Coal and Wind Renewable Energy Sources.
- Economic dispatch Solutions for power grid energy management systems in national or regional control centers' decision-making.

- Apply the developed economic dispatch algorithm for distributed and parallel power system network optimization.
- Apply the developed optimization methods in postgraduate research and educational courses
- Develop a framework for Smart Grids by solving the combined wind-coal economic dispatch problem.

## 7.5 Future research

Potential advances can be listed as follows:

- Applying the created techniques, algorithms, and programs for demand response, electric vehicles (EVs), hydro, wind, solar, storage, and pumped storage systems.
- Application development, programming, and solution implementation to the multi-area dispatch problem using a Cluster of Computers.
- More heuristic algorithms can be implemented and tested more efficiently to achieve the global optimum of the WTEED problems.
- Proposed algorithms can be implemented in a looped optimisation of the dispatch problem using CC and RTDS with a real-time lab open to network connectivity.
- Validate the developed application on larger IEEE systems.

## 7.6 Publications

Mbangeni, L., & Krishnamurthy, S. (2025). A Lagrange-Based Multi-Objective Framework for Wind-Thermal Economic Emission Dispatch. MDPI(*Processes)*, *13*(9), 2814. https://doi.org/10.3390/pr13092814

Mbangeni, L., & Krishnamurthy, S. (2026). Performance Assessment of Particle Swarm Optimization for Wind Thermal_Economic_Emission_Dispatc submit to Optimisation and Engineering, Springer, Journal (Under Review) in February 2026

Mbangeni, L., & Krishnamurthy, S. (2026). Implementation of the Lagrange decomposition Method for solution of the Multi-Area Wind-Thermal Economic Emission Dispatch (MAWTEED) problem submitted to Results in Control and Optimization Journal(Under Review) in February 2026.

Mbangeni, L., & Krishnamurthy, S. (2026). Multi-area wind-thermal economic dispatch problem using the PSO Method, Anticipated to submit to Power and Energy Conversion Journal in February 2026.

## 7.7 Conclusion

The goals and objectives of the thesis, as well as the deliverables for developing techniques, algorithms, and software for single and multi-area wind-thermal economic emission dispatch problems, are outlined in this chapter. The chapter offers recommendations on customising software, algorithms, and the proposed dispatch problem technique for use in utility applications. A list of the author's publications and the direction of future study is provided.

## REFERENCES

Abdelaziz, A. Y., Ali, E. S., & Abd Elazim, S. M. (2016). Implementation of flower pollination algorithm for solving economic load dispatch and combined economic emission dispatch problems in power systems. *Energy*, *101*, 506–518. https://doi.org/10.1016/j.energy.2016.02.041

Ahmed, I., Rehan, M., Basit, A., Malik, S. H., Alvi, U.-E.-H., & Hong, K.-S. (2022). Multi-area economic emission dispatch for large-scale multi-fueled power plants contemplating inter-connected grid tie-lines power flow limitations. *Energy*, *261*, 125178. https://doi.org/10.1016/j.energy.2022.125178

Alham, M. H., Elshahed, M., Ibrahim, D. K., & Abo El Zahab, E. E. D. (2016). A dynamic economic emission dispatch considering wind power uncertainty incorporating energy storage system and demand side management. *Renewable Energy*, *96*, 800–811. https://doi.org/10.1016/j.renene.2016.05.012

Alli, K. (2024). An Extended Multi-objective Multi-area Emission Economic Dispatch Problem Integrating Random Wind Power Units Using a Semi-definite Programming Method. *The West Indian Journal of Engineering*, *Vol.47*(No.1), 46–55.

Arabali, A., Ghofrani, M., Etezadi-Amoli, M., & Fadali, M. S. (2014). Stochastic Performance Assessment and Sizing for a Hybrid Power System of Solar/Wind/Energy Storage. *IEEE Transactions on Sustainable Energy*, *5*(2), 363–371. https://doi.org/10.1109/TSTE.2013.2288083

Arriagada, E., López, E., López, M., Blasco-Gimenez, R., Roa, C., & Poloujadoff, M. (2015). A probabilistic economic dispatch model and methodology considering renewable energy, demand and generator uncertainties. *Electric Power Systems Research*, *121*, 325–332. https://doi.org/10.1016/j.epsr.2014.11.018

Atia, R., & Yamada, N. (2016). Distributed Renewable Generation and Storage System Sizing Based on Smart Dispatch of Microgrids. *Energies*, *9*(3), 176. https://doi.org/10.3390/en9030176

Azeem, M., Malik, T. N., Muqeet, H. A., Hussain, M. M., Ali, A., Khan, B., & Rehman, A. U. (2023). Combined Economic Emission Dispatch in Presence of Renewable

Energy Resources Using CISSA in a Smart Grid Environment. *Electronics*, *12*(3), 715. https://doi.org/10.3390/electronics12030715

Azizipanah-Abarghooee, R., Niknam, T., Bina, M. A., & Zare, M. (2015). Coordination of combined heat and power-thermal-wind-photovoltaic units in economic load dispatch using chance-constrained and jointly distributed random variables methods. *Energy*, *79*, 50–67. https://doi.org/10.1016/j.energy.2014.10.024

Azizipanah-Abarghooee, R., Niknam, T., Roosta, A., Malekpour, A. R., & Zare, M. (2012). Probabilistic multiobjective wind-thermal economic emission dispatch based on point estimated method. *Energy*, *37*(1), 322–335. https://doi.org/10.1016/j.energy.2011.11.023

Basu, M. (2011a). Economic environmental dispatch using multi-objective differential evolution. *Applied Soft Computing*, *11*(2), 2845–2853. https://doi.org/10.1016/j.asoc.2010.11.014

Basu, M. (2011b). Economic environmental dispatch using multi-objective differential evolution. *Applied Soft Computing*, *11*(2), 2845–2853. https://doi.org/10.1016/j.asoc.2010.11.014

Basu, M. (2013). Artificial bee colony optimization for multi-area economic dispatch. *International Journal of Electrical Power & Energy Systems*, *49*, 181–187. https://doi.org/10.1016/j.ijepes.2013.01.004

Basu, M. (2014). Teaching–learning-based optimization algorithm for multi-area economic dispatch. *Energy*, *68*, 21–28. https://doi.org/10.1016/j.energy.2014.02.064

Basu, M. (2019a). Multi-area dynamic economic emission dispatch of hydro-wind-thermal power system. *Renewable Energy Focus*, *28*, 11–35. https://doi.org/10.1016/j.ref.2018.09.007

Basu, M. (2019b). Multi-region dynamic economic dispatch of solar–wind–hydro–thermal power system incorporating pumped hydro energy storage. *Engineering Applications of Artificial Intelligence*, *86*, 182–196. https://doi.org/10.1016/j.engappai.2019.09.001

Behera, S., Behera, S., Barisal, A. K., & Sahu, P. (2021). Dynamic economic emission dispatch of thermal-wind-solar system with constriction factor-based particle swarm optimization algorithm. *World Journal of Engineering*, *18*(2), 217–227. https://doi.org/10.1108/WJE-07-2020-0327

Belmili, H., Haddadi, M., Bacha, S., Almi, M. F., & Bendib, B. (2014). Sizing stand-alone photovoltaic–wind hybrid system: Techno-economic analysis and optimization. *Renewable and Sustainable Energy Reviews*, *30*, 821–832. https://doi.org/10.1016/j.rser.2013.11.011

Bhattacharya, A., & Chattopadhyay, P. K. (2010). Biogeography-Based Optimization for Different Economic Load Dispatch Problems. *IEEE Transactions on Power Systems*, *25*(2), 1064–1077. https://doi.org/10.1109/TPWRS.2009.2034525

Bilil, H., Aniba, G., & Maaroufi, M. (2014). Probabilistic Economic Emission Dispatch Optimization of Multi-sources Power System. *Energy Procedia*, *50*, 789–796. https://doi.org/10.1016/j.egypro.2014.06.097

Boqiang, R., & Chuanwen, J. (2009). A review on the economic dispatch and risk management considering wind power in the power market. *Renewable and Sustainable Energy Reviews*, *13*(8), 2169–2174. https://doi.org/10.1016/j.rser.2009.01.013

Cao, J., Du, W., & Wang, H. F. (2016). Weather-Based Optimal Power Flow With Wind Farms Integration. *IEEE Transactions on Power Systems*, *31*(4), 3073–3081. https://doi.org/10.1109/TPWRS.2015.2488662

Chaudhary, V., Dubey, H. M., Pandit, M., & Bansal, J. C. (2020). Multi-area economic dispatch with stochastic wind power using Salp Swarm Algorithm. *Array*, *8*, 100044. https://doi.org/10.1016/j.array.2020.100044

Chen, C.-L., Lee, T.-Y., & Jan, R.-M. (2006). Optimal wind-thermal coordination dispatch in isolated power systems with large integration of wind capacity. *Energy Conversion and Management*, *47*(18–19), 3456–3472. https://doi.org/10.1016/j.enconman.2005.12.016

Chen, F., Zhou, J., Wang, C., Li, C., & Lu, P. (2017). A modified gravitational search algorithm based on a non-dominated sorting genetic approach for hydro-thermal-wind economic emission dispatching. *Energy*, *121*, 276–291. https://doi.org/10.1016/j.energy.2017.01.010

Chopra, N., Brar, Y. S., & Dhillon, J. S. (2018). Multiarea Economic-Emission Dispatch Using Simplex Based Particle Swarm Optimization Method. *Asian Journal of Electrical Sciences*, *7*(1), 29–36. https://doi.org/10.51983/ajes-2018.7.1.2058

Damodaran, S. K., & Kumar, T. K. S. (2014). Combined Economic and Emission Dispatch Using a Classical Technique. *IFAC Proceedings Volumes*, *47*(1), 1049–1053. https://doi.org/10.3182/20140313-3-IN-3024.00049

Damodaran, S., & Sunil Kumar, T. (2018). Hydro-Thermal-Wind Generation Scheduling Considering Economic and Environmental Factors Using Heuristic Algorithms. *Energies*, *11*(2), 353. https://doi.org/10.3390/en11020353

Diaf, S., Belhamel, M., Haddadi, M., & Louche, A. (2008). Technical and economic assessment of hybrid photovoltaic/wind system with battery storage in Corsica island. *Energy Policy*, *36*(2), 743–754. https://doi.org/10.1016/j.enpol.2007.10.028

Dubey, H. M., Pandit, M., & Panigrahi, B. K. (2015). Hybrid flower pollination algorithm with time-varying fuzzy selection mechanism for wind integrated multi-objective dynamic economic dispatch. *Renewable Energy*, *83*, 188–202. https://doi.org/10.1016/j.renene.2015.04.034

Dufo-López, R., & Bernal-Agustín, J. L. (2005). Design and control strategies of PV-Diesel systems using genetic algorithms. *Solar Energy*, *79*(1), 33–46. https://doi.org/10.1016/j.solener.2004.10.004

Dufo-López, R., & Bernal-Agustín, J. L. (2008). Multi-objective design of PV–wind–diesel–hydrogen–battery systems. *Renewable Energy*, *33*(12), 2559–2572. https://doi.org/10.1016/j.renene.2008.02.027

Dufo-López, R., Bernal-Agustín, J. L., & Contreras, J. (2007). Optimization of control strategies for stand-alone renewable energy systems with hydrogen storage. *Renewable Energy*, *32*(7), 1102–1126. https://doi.org/10.1016/j.renene.2006.04.013

Elattar, E. E. (2018). Modified harmony search algorithm for combined economic emission dispatch of microgrid incorporating renewable sources. *Energy*, *159*, 496–507. https://doi.org/10.1016/j.energy.2018.06.137

ElDesouky, A. A. (2014). Security constrained generation scheduling for grids incorporating wind, photovoltaic and thermal power. *Electric Power Systems Research*, *116*, 284–292. https://doi.org/10.1016/j.epsr.2014.06.017

Fesanghary, M., & Ardehali, M. M. (2009). A novel meta-heuristic optimization methodology for solving various types of economic dispatch problem. *Energy*, *34*(6), 757–766. https://doi.org/10.1016/j.energy.2009.02.007

Ghasemi, A., Gheydi, M., Golkar, M. J., & Eslami, M. (2016). Modeling of Wind/Environment/Economic Dispatch in power system and solving via an online learning meta-heuristic method. *Applied Soft Computing*, *43*, 454–468. https://doi.org/10.1016/j.asoc.2016.02.046

Ghasemi, M., Aghaei, J., Akbari, E., Ghavidel, S., & Li, L. (2016). A differential evolution particle swarm optimizer for various types of multi-area economic dispatch problems. *Energy*, *107*, 182–195. https://doi.org/10.1016/j.energy.2016.04.002

Giannakoudis, G., Papadopoulos, A. I., Seferlis, P., & Voutetakis, S. (2010). Optimum design and operation under uncertainty of power systems using renewable energy sources and hydrogen storage. *International Journal of Hydrogen Energy*, *35*(3), 872–891. https://doi.org/10.1016/j.ijhydene.2009.11.044

Guo, L., Liu, W., Li, X., Liu, Y., Jiao, B., Wang, W., Wang, C., & Li, F. (2014). Energy Management System for Stand-Alone Wind-Powered-Desalination Microgrid. *IEEE Transactions on Smart Grid*, 1–1. https://doi.org/10.1109/TSG.2014.2377374

Hetzer, J., Yu, D. C., & Bhattarai, K. (2008). An Economic Dispatch Model Incorporating Wind Power. *IEEE Transactions on Energy Conversion*, *23*(2), 603–611. https://doi.org/10.1109/TEC.2007.914171

Hu, F., Hughes, K. J., Ingham, D. B., Ma, L., & Pourkashanian, M. (2019). Dynamic economic and emission dispatch model considering wind power under Energy Market Reform: A case study. *International Journal of Electrical Power & Energy Systems*, *110*, 184–196. https://doi.org/10.1016/j.ijepes.2019.03.004

Huang, W.-T., Yao, K.-C., Chen, M.-K., Wang, F.-Y., Zhu, C.-H., Chang, Y.-R., Lee, Y.-D., & Ho, Y.-H. (2018). Derivation and Application of a New Transmission Loss Formula for Power System Economic Dispatch. *Energies*, *11*(2), 417. https://doi.org/10.3390/en11020417

Jadhav, H. T., & Roy, R. (2013a). Gbest guided artificial bee colony algorithm for environmental/economic dispatch considering wind power. *Expert Systems with Applications*, *40*(16), 6385–6399. https://doi.org/10.1016/j.eswa.2013.05.048

Jadhav, H. T., & Roy, R. (2013b). Gbest guided artificial bee colony algorithm for environmental/economic dispatch considering wind power. *Expert Systems with Applications*, *40*(16), 6385–6399. https://doi.org/10.1016/j.eswa.2013.05.048

Jadoun, V. K., Gupta, N., Niazi, K. R., & Swarnkar, A. (2015). Multi-area economic dispatch with reserve sharing using dynamically controlled particle swarm optimization. *International Journal of Electrical Power & Energy Systems*, *73*, 743–756. https://doi.org/10.1016/j.ijepes.2015.06.008

Jayabarathi, T., Jayaprakash, K., Jeyakumar, D. N., & Raghunathan, T. (2005). Evolutionary programming techniques for different kinds of economic dispatch problems. *Electric Power Systems Research*, *73*(2), 169–176. https://doi.org/10.1016/j.epsr.2004.08.001

Jeyakumar, D. N., Jayabarathi, T., & Raghunathan, T. (2006). Particle swarm optimization for various types of economic dispatch problems. *International Journal of Electrical Power & Energy Systems*, *28*(1), 36–42. https://doi.org/10.1016/j.ijepes.2005.09.004

Jiang, S., Ji, Z., & Wang, Y. (2015). A novel gravitational acceleration enhanced particle swarm optimization algorithm for wind–thermal economic emission dispatch problem considering wind power availability. *International Journal of Electrical Power & Energy Systems*, *73*, 1035–1050. https://doi.org/10.1016/j.ijepes.2015.06.014

Jiang, W., Yan, Z., & Hu, Z. (2011). A Novel Improved Particle Swarm Optimization Approach for Dynamic Economic Dispatch Incorporating Wind Power. *Electric Power Components and Systems*, *39*(5), 461–477. https://doi.org/10.1080/15325008.2010.528536

Jin, J., Wen, Q., Zhang, X., Cheng, S., & Guo, X. (2021). Economic Emission Dispatch for Wind Power Integrated System with Carbon Trading Mechanism. *Energies*, *14*(7), 1870. https://doi.org/10.3390/en14071870

Kaabeche, A., Belhamel, M., & Ibtiouen, R. (2011). Sizing optimization of grid-independent hybrid photovoltaic/wind power generation system. *Energy*, *36*(2), 1214–1222. https://doi.org/10.1016/j.energy.2010.11.024

Kamjoo, A., Maheri, A., Dizqah, A. M., & Putrus, G. A. (2016). Multi-objective design under uncertainties of hybrid renewable energy system using NSGA-II and chance constrained programming. *International Journal of Electrical Power & Energy Systems*, *74*, 187–194. https://doi.org/10.1016/j.ijepes.2015.07.007

Kashefi Kaviani, A., Riahy, G. H., & Kouhsari, Sh. M. (2009). Optimal design of a reliable hydrogen-based stand-alone wind/PV generating system, considering component outages. *Renewable Energy*, *34*(11), 2380–2390. https://doi.org/10.1016/j.renene.2009.03.020

Kothari, D. P., & Dhillon, J. S. (2012). *Power system optimization* (2nd ed). PHI Learning Private Ltd.

Krishnamurthy, S., & Tzoneva, R. (2012). Comparison of the Lagrange's and Particle Swarm Optimisation solutions of an Economic Emission Dispatch problem with transmission constraints. *2012 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 1–8. https://doi.org/10.1109/PEDES.2012.6484295

Krishnamurthy, S., & Tzoneva, R. (2016). Decomposition-Coordinating Method for Parallel Solution of a Multi-area Combined Economic Emission Dispatch Problem. *International Journal of Electrical and Computer Engineering (IJECE)*, *6*(5), 2048. https://doi.org/10.11591/ijece.v6i5.10259

Krishnamurthy, S., Tzoneva, R., & Apostolov, A. (2017). Method for a Parallel Solution of a Combined Economic Emission Dispatch Problem. *Electric Power Components and Systems*, *45*(4), 393–409. https://doi.org/10.1080/15325008.2016.1265614

Krishnasamy, U., & Nanjundappan, D. (2016). Hybrid weighted probabilistic neural network and biogeography based optimization for dynamic economic dispatch of integrated multiple-fuel and wind power plants. *International Journal of Electrical Power & Energy Systems*, *77*, 385–394. https://doi.org/10.1016/j.ijepes.2015.11.022

Kumari, M., & Pal, K. (2022). Dynamic Combined Economic Emission Dispatch Including Wind Generators by Real Coded Genetic Algorithm: *International Journal of Applied Metaheuristic Computing*, *13*(1), 1–23. https://doi.org/10.4018/IJAMC.292510

Kuo, C.-C. (2010). Wind energy dispatch considering environmental and economic factors. *Renewable Energy*, *35*(10), 2217–2227. https://doi.org/10.1016/j.renene.2010.02.023

Kusakana, K. (2019). Optimal electricity cost minimization of a grid-interactive Pumped Hydro Storage using ground water in a dynamic electricity pricing environment. *Energy Reports*, *5*, 159–169. https://doi.org/10.1016/j.egyr.2019.01.004

Lalhmachhuana, R., Deb, S., Datta, S., Singh, Ksh. R., Cali, U., & Ustun, T. S. (2024). Multi-objective-based economic and emission dispatch with integration of wind energy sources using different optimization algorithms. *Frontiers in Energy Research*, *12*, 1421212. https://doi.org/10.3389/fenrg.2024.1421212

Lee, T.-Y. (2007). Optimal Spinning Reserve for a Wind-Thermal Power System Using EIPSO. *IEEE Transactions on Power Systems*, *22*(4), 1612–1621. https://doi.org/10.1109/TPWRS.2007.907519

Leon-Garcia, A. (2008). *Probability, statistics, and random processes for electrical engineering* (3. ed). Pearson/Prentice Hall.

Li, J., Xu, L., Zhang, Y., Kou, Y., Liang, W., Bieerke, A., & Yuan, Z. (2024). Economic Dispatch of Integrated Energy Systems Considering Wind–Photovoltaic Uncertainty and Efficient Utilization of Electrolyzer Thermal Energy. *Processes*, *12*(8), 1627. https://doi.org/10.3390/pr12081627

Li, X., & Jiang, C. (2011). Short-Term Operation Model and Risk Management for Wind Power Penetrated System in Electricity Market. *IEEE Transactions on Power Systems*, *26*(2), 932–939. https://doi.org/10.1109/TPWRS.2010.2070882

Li, Z., Wu, W., Zhang, B., Sun, H., & Guo, Q. (2013). Dynamic Economic Dispatch Using Lagrangian Relaxation With Multiplier Updates Based on a Quasi-Newton

Method. *IEEE Transactions on Power Systems*, *28*(4), 4516–4527. https://doi.org/10.1109/TPWRS.2013.2267057

Liang, R.-H., & Liao, J.-H. (2007). A Fuzzy-Optimization Approach for Generation Scheduling With Wind and Solar Energy Systems. *IEEE Transactions on Power Systems*, *22*(4), 1665–1674. https://doi.org/10.1109/TPWRS.2007.907527

Liang, R.-H., Wang, J.-C., Chen, Y.-T., & Tseng, W.-T. (2015). An enhanced firefly algorithm to multi-objective optimal active/reactive power dispatch with uncertainties consideration. *International Journal of Electrical Power & Energy Systems*, *64*, 1088–1097. https://doi.org/10.1016/j.ijepes.2014.09.008

Liu, G., Zhu, Y. L., & Jiang, W. (2018). Wind-thermal dynamic economic emission dispatch with a hybrid multi-objective algorithm based on wind speed statistical analysis. *IET Generation, Transmission & Distribution*, *12*(17), 3972–3984. https://doi.org/10.1049/iet-gtd.2018.5364

Liu, X. (2010). Economic Load Dispatch Constrained by Wind Power Availability: A Wait-and-See Approach. *IEEE Transactions on Smart Grid*, *1*(3), 347–355. https://doi.org/10.1109/TSG.2010.2057458

Liu, X., & Xu, W. (2010a). Economic Load Dispatch Constrained by Wind Power Availability: A Here-and-Now Approach. *IEEE Transactions on Sustainable Energy*, *1*(1), 2–9. https://doi.org/10.1109/TSTE.2010.2044817

Liu, X., & Xu, W. (2010b). Economic Load Dispatch Constrained by Wind Power Availability: A Here-and-Now Approach. *IEEE Transactions on Sustainable Energy*, *1*(1), 2–9. https://doi.org/10.1109/TSTE.2010.2044817

Liu, Z.-F., Li, L.-L., Liu, Y.-W., Liu, J.-Q., Li, H.-Y., & Shen, Q. (2021). Dynamic economic emission dispatch considering renewable energy generation: A novel multi-objective optimization approach. *Energy*, *235*, 121407. https://doi.org/10.1016/j.energy.2021.121407

Lu, B., & Shahidehpour, M. (2005). Short-Term Scheduling of Battery in a Grid-Connected PV/Battery System. *IEEE Transactions on Power Systems*, *20*(2), 1053–1061. https://doi.org/10.1109/TPWRS.2005.846060

Lu, C.-L., Chen, C.-L., Hwang, D.-S., & Cheng, Y.-T. (2008). Effects of wind energy supplied by independent power producers on the generation dispatch of electric power utilities. *International Journal of Electrical Power & Energy Systems*, *30*(9), 553–561. https://doi.org/10.1016/j.ijepes.2008.08.002

Maleki, A., & Askarzadeh, A. (2014). Comparative study of artificial intelligence techniques for sizing of a hydrogen-based stand-alone photovoltaic/wind hybrid system. *International Journal of Hydrogen Energy*, *39*(19), 9973–9984. https://doi.org/10.1016/j.ijhydene.2014.04.147

Malheiro, A., Castro, P. M., Lima, R. M., & Estanqueiro, A. (2015). Integrated sizing and scheduling of wind/PV/diesel/battery isolated systems. *Renewable Energy*, *83*, 646–657. https://doi.org/10.1016/j.renene.2015.04.066

Manoharan, P. S., Kannan, P. S., Baskar, S., & Willjuice Iruthayarajan, M. (2009). Evolutionary algorithm solution and KKT based optimality verification to multi-area economic dispatch. *International Journal of Electrical Power & Energy Systems*, *31*(7–8), 365–373. https://doi.org/10.1016/j.ijepes.2009.03.010

Manoharan, P. S., Kannan, P. S., & Ramanathan, V. (2009). A Novel EP Approach for Multi-area Economic Dispatch with Multiple Fuel Options. *Turkish Journal of Electrical Engineering and Computer Sciences*. https://doi.org/10.3906/elk-0705-10

Mao, H., Deng, Q., Zhang, Z., & Yang, X. (2026). Optimized economic scheduling of demand response in integrated energy systems considering dynamic energy efficiency and dynamic carbon trading. *Scientific Reports*, *16*(1), 1350. https://doi.org/10.1038/s41598-025-33497-3

Masters, G. M. (2004). *Renewable and Efficient Electric Power Systems* (1st ed.). Wiley. https://doi.org/10.1002/0471668826

Meng, Q., He, Y., Hussain, S., Lu, J., & Guerrero, J. M. (2025). Day-ahead economic dispatch of wind-integrated microgrids using coordinated energy storage and hybrid demand response strategies. *Scientific Reports*, *15*(1), 26579. https://doi.org/10.1038/s41598-025-11561-2

Meyyappan, U., & Pandu, K. D. R. (2015). Wavelet Neural Network-based Wind-power Forecasting in Economic Dispatch: A Differential Evolution, Bacterial Foraging

Technology, and Primal Dual-interior Point Approach. *Electric Power Components and Systems*, *43*(13), 1467–1477. https://doi.org/10.1080/15325008.2015.1043600

Ming Chen, Y., & Wang, W.-S. (2010). A particle swarm approach to solve environmental/economic dispatch problem. *International Journal of Industrial Engineering Computations*, *1*(2), 157–172. https://doi.org/10.5267/j.ijiec.2010.02.005

Mishra, S., & Shaik, A. G. (2024a). Solving bi-objective economic-emission load dispatch of diesel-wind-solar microgrid using African vulture optimization algorithm. *Heliyon*, *10*(3), e24993. https://doi.org/10.1016/j.heliyon.2024.e24993

Mishra, S., & Shaik, A. G. (2024b). Solving bi-objective economic-emission load dispatch of diesel-wind-solar microgrid using African vulture optimization algorithm. *Heliyon*, *10*(3), e24993. https://doi.org/10.1016/j.heliyon.2024.e24993

Mondal, S., Bhattacharya, A., & Nee Dey, S. H. (2013a). Multi-objective economic emission load dispatch solution using gravitational search algorithm and considering wind power penetration. *International Journal of Electrical Power & Energy Systems*, *44*(1), 282–292. https://doi.org/10.1016/j.ijepes.2012.06.049

Mondal, S., Bhattacharya, A., & Nee Dey, S. H. (2013b). Multi-objective economic emission load dispatch solution using gravitational search algorithm and considering wind power penetration. *International Journal of Electrical Power & Energy Systems*, *44*(1), 282–292. https://doi.org/10.1016/j.ijepes.2012.06.049

Morales, J. M., Conejo, A. J., & Perez-Ruiz, J. (2009). Economic valuation of reserves in power systems with high penetration of wind power. *2009 IEEE Power & Energy Society General Meeting*, 1–1. https://doi.org/10.1109/PES.2009.5260229

Nagarajan, K., Rajagopalan, A., Angalaeswari, S., Natrayan, L., & Mammo, W. D. (2022). Combined Economic Emission Dispatch of Microgrid with the Incorporation of Renewable Energy Sources Using Improved Mayfly Optimization Algorithm. *Computational Intelligence and Neuroscience*, *2022*, 1–22. https://doi.org/10.1155/2022/6461690

Ogunjuyigbe, A. S. O., Ayodele, T. R., & Akinola, O. A. (2016). Optimal allocation and sizing of PV/Wind/Split-diesel/Battery hybrid energy system for minimizing life cycle cost, carbon emission and dump energy of remote residential building. *Applied Energy*, *171*, 153–171. https://doi.org/10.1016/j.apenergy.2016.03.051

Pandit, M., Chaudhary, V., Dubey, H. M., & Panigrahi, B. K. (2015a). Multi-period wind integrated optimal dispatch using series PSO-DE with time-varying Gaussian membership function based fuzzy selection. *International Journal of Electrical Power & Energy Systems*, *73*, 259–272. https://doi.org/10.1016/j.ijepes.2015.05.017

Pandit, M., Chaudhary, V., Dubey, H. M., & Panigrahi, B. K. (2015b). Multi-period wind integrated optimal dispatch using series PSO-DE with time-varying Gaussian membership function based fuzzy selection. *International Journal of Electrical Power & Energy Systems*, *73*, 259–272. https://doi.org/10.1016/j.ijepes.2015.05.017

Pandit, M., Srivastava, L., & Sharma, M. (2015). Environmental economic dispatch in multi-area power system employing improved differential evolution with fuzzy selection. *Applied Soft Computing*, *28*, 498–510. https://doi.org/10.1016/j.asoc.2014.12.027

Patel, M. R. (1999). *Wind and solar power systems*. CRC Press.

Peng, C., Sun, H., Guo, J., & Liu, G. (2012a). Dynamic economic dispatch for wind-thermal power system using a novel bi-population chaotic differential evolution algorithm. *International Journal of Electrical Power & Energy Systems*, *42*(1), 119–126. https://doi.org/10.1016/j.ijepes.2012.03.012

Peng, C., Sun, H., Guo, J., & Liu, G. (2012b). Dynamic economic dispatch for wind-thermal power system using a novel bi-population chaotic differential evolution algorithm. *International Journal of Electrical Power & Energy Systems*, *42*(1), 119–126. https://doi.org/10.1016/j.ijepes.2012.03.012

Plathottam, S. J., & Salehfar, H. (2015). Unbiased economic dispatch in control areas with conventional and renewable generation sources. *Electric Power Systems Research*, *119*, 313–321. https://doi.org/10.1016/j.epsr.2014.09.025

Qiao, B., Feng, Z., Zhu, Z., Yan, L., Qu, B., Chai, X., & Huan, J. (2025). Multi-objective dynamic economic emission dispatch considering the risk of wind power-electric vehicle interaction using teaching-learning-based optimization algorithm. *Swarm and Evolutionary Computation*, *98*, 102096. https://doi.org/10.1016/j.swevo.2025.102096

Qu, B. Y., Liang, J. J., Zhu, Y. S., Wang, Z. Y., & Suganthan, P. N. (2016). Economic emission dispatch problems with stochastic wind power using summation based multi-objective evolutionary algorithm. *Information Sciences*, *351*, 48–66. https://doi.org/10.1016/j.ins.2016.01.081

Reddy, S. S., Panigrahi, B. K., Kundu, R., Mukherjee, R., & Debchoudhury, S. (2013). Energy and spinning reserve scheduling for a wind-thermal power system using CMA-ES with mean learning technique. *International Journal of Electrical Power & Energy Systems*, *53*, 113–122. https://doi.org/10.1016/j.ijepes.2013.03.032

Sakthivel, V. P., Nirmal Kumar, S., & Sathya, P. D. (2024). Multi-Objective Dynamic Economic and Emission Generation Scheduling of Coordinated Power System Considering Renewable Energy Sources and Pumped Storage Hydro Plants via Fuzzy Surrogate-Assisted Coronavirus Herd Immunity Optimization. *Electric Power Components and Systems*, 1–30. https://doi.org/10.1080/15325008.2024.2343397

Salkuti, S. R. (2021). Solving combined economic emission dispatch problem in wind integrated power systems. *Indonesian Journal of Electrical Engineering and Computer Science*, *21*(2), 635. https://doi.org/10.11591/ijeecs.v21.i2.pp635-641

Secui, D. C. (2015). The chaotic global best artificial bee colony algorithm for the multi-area economic/emission dispatch. *Energy*, *93*, 2518–2545. https://doi.org/10.1016/j.energy.2015.10.012

Secui, D. C., Hora, C., Bendea, C., Secui, M. L., Bendea, G., & Dan, F. C. (2024a). Modified Social Group Optimization to Solve the Problem of Economic Emission Dispatch with the Incorporation of Wind Power. *Sustainability*, *16*(1), 397. https://doi.org/10.3390/su16010397

Secui, D. C., Hora, C., Bendea, C., Secui, M. L., Bendea, G., & Dan, F. C. (2024b). Modified Social Group Optimization to Solve the Problem of Economic Emission Dispatch with the Incorporation of Wind Power. *Sustainability*, *16*(1), 397. https://doi.org/10.3390/su16010397

Sharma, M., Pandit, M., & Srivastava, L. (2011). Reserve constrained multi-area economic dispatch employing differential evolution with time-varying mutation.

*International Journal of Electrical Power & Energy Systems*, *33*(3), 753–766. https://doi.org/10.1016/j.ijepes.2010.12.033

Shayeghi, H., Moradzadeh, M., Hashemi, Y., Saif, M., & Vandevelde, L. (2015). Wind-PV-storage optimal environomic design using multi-objective Artificial Bee Colony. *2015 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, 1–5. https://doi.org/10.1109/APPEEC.2015.7381057

Som, T., & Chakraborty, N. (2014). Studies on Economic Feasibility of an Autonomous Power Delivery System Utilizing Alternative Hybrid Distributed Energy Resources. *IEEE Transactions on Power Systems*, *29*(1), 172–181. https://doi.org/10.1109/TPWRS.2013.2280214

Somasundaram, P., & Swaroopan, N. M. J. (2011). Fuzzified Particle Swarm Optimization Algorithm for Multi-area Security Constrained Economic Dispatch. *Electric Power Components and Systems*, *39*(10), 979–990. https://doi.org/10.1080/15325008.2011.552094

Soroudi, A., & Rabiee, A. (2013). Optimal multi-area generation schedule considering renewable resources mix: A real-time approach. *IET Generation, Transmission & Distribution*, *7*(9), 1011–1026. https://doi.org/10.1049/iet-gtd.2012.0735

Theo, W. L., Lim, J. S., Ho, W. S., Hashim, H., & Lee, C. T. (2017). Review of distributed generation (DG) system planning and optimisation techniques: Comparison of numerical and mathematical modelling methods. *Renewable and Sustainable Energy Reviews*, *67*, 531–573. https://doi.org/10.1016/j.rser.2016.09.063

Wang, C., & Shahidehpour, S. M. (1992). A decomposition approach to nonlinear multi-area generation scheduling with tie-line constraints using expert systems. *IEEE Transactions on Power Systems*, *7*(4), 1409–1418. https://doi.org/10.1109/59.207362

Wang, L., & Singh, C. (2007). Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization algorithm. *Electric Power Systems Research*, *77*(12), 1654–1664. https://doi.org/10.1016/j.epsr.2006.11.012

Wang, L., & Singh, C. (2009). Reserve-constrained multiarea environmental/economic dispatch based on particle swarm optimization with local search. *Engineering*

*Applications of Artificial Intelligence*, *22*(2), 298–307. https://doi.org/10.1016/j.engappai.2008.07.007

Wang, Y., Xiong, G., Xu, S., & Suganthan, P. N. (2025). Dynamic economic dispatch of multi-area wind-solar-thermal power systems with fractional order comprehensive learning differential evolution. *Energy*, *326*, 136233. https://doi.org/10.1016/j.energy.2025.136233

Xia, X., & Elaiw, A. M. (2010). Optimal dynamic economic dispatch of generation: A review. *Electric Power Systems Research*, *80*(8), 975–986. https://doi.org/10.1016/j.epsr.2009.12.012

Xian Liu. (2012). An Improved Interpolation Method for Wind Power Curves. *IEEE Transactions on Sustainable Energy*, *3*(3), 528–534. https://doi.org/10.1109/TSTE.2012.2191582

Xian Liu, & Xu, W. (2010). Minimum Emission Dispatch Constrained by Stochastic Wind Power Availability and Cost. *IEEE Transactions on Power Systems*, *25*(3), 1705–1713. https://doi.org/10.1109/TPWRS.2010.2042085

Xiong, G., Liu, Q., Wang, Y., & Fu, X. (2025). Power system economic emission dispatch considering uncertainties of wind, solar, and small runoff hydropower via a hybrid multi-objective optimization algorithm. *Expert Systems with Applications*, *278*, 127375. https://doi.org/10.1016/j.eswa.2025.127375

Yan, L., Zhu, Z., Kang, X., Qu, B., Qiao, B., Huan, J., & Chai, X. (2022). Multi-Objective Dynamic Economic Emission Dispatch with Electric Vehicle–Wind Power Interaction Based on a Self-Adaptive Multiple-Learning Harmony-Search Algorithm. *Energies*, *15*(14), 4942. https://doi.org/10.3390/en15144942

Zhang, C., Wu, W., & Chen, S. (2019). Combined Economic and Emission Dispatch Problem of Wind-Thermal Power System Using Gravitational Particle Swarm Optimization Algorithm. *Mathematical Problems in Engineering*, *2019*(1), 5679361. https://doi.org/10.1155/2019/5679361

Zhou, W., Peng, Y., & Sun, H. (2011). Optimal wind-thermal coordination dispatch based on risk reserve constraints. *European Transactions on Electrical Power*, *21*(1), 740–756. https://doi.org/10.1002/etep.474

Zhu, J. (2015). *OPTIMIZATION OF POWER SYSTEM OPERATION* (1st ed.). Wiley. https://doi.org/10.1002/9781118887004

Zhu, Y., Wang, J., & Qu, B. (2014). Multi-objective economic emission dispatch considering wind power using evolutionary algorithm based on decomposition. *International Journal of Electrical Power & Energy Systems*, *63*, 434–445. https://doi.org/10.1016/j.ijepes.2014.06.027

Zou, D., Gong, D., & Ouyang, H. (2023). The dynamic economic emission dispatch of the combined heat and power system integrated with a wind farm and a photovoltaic plant. *Applied Energy*, *351*, 121890. https://doi.org/10.1016/j.apenergy.2023.121890

# APPENDICES

## APPENDIX A: DEVELOPED MATLAB PROGRAM FOR A SINGLE AREA WTEED PROBLEM USING LAGRANGE'S ALGORITHM

### APPENDIX A1: MATLAB script file – WTEED_case6units.m

```
% M-File : WTEED_cases6units.m
% IEEE 30 bus SIX GENERATOR SYSTEM
% ==================M-File Description===============================
% The M-File is used to solve WTEED problem using Lagrange's Algorithm
clearvars, clc
tic;
global lambda PTmin PTmax n a b c d e f B B01 B00 h;
propmt='Enter the Power demand value (in MW)= ';
PD = input(propmt);
%===========Fuel cost coefficients========================================
a=[10 10 20 10 20 10];
b=[200 150 180 100 180 150];
c=[100 120 40 60 40 100];
%%==============Emission  coefficients===================================
d=[4.091 2.543 4.258 5.326 4.258 6.131];
e=[-5.554 -6.047 -5.094 -3.55 -5.094 -5.555];
f=[6.49 5.638 4.586 3.38 4.586 5.151];
Wmin = 1;
Wmax = 200;
nW = 1;
%================= Transmission loss coefficients==========================
B00=[0.0014];
B01=[0.010731 1.7704 -4.0645 3.8453 1.3832 5.5503]*10.^(-3);
B=[ 0.0218 0.0107 -0.00036 -0.0011 0.00055 0.0033
0.0107 0.01704 -0.0001 -0.00179 0.00026 0.0028
-0.0004 -0.0002 0.02459 -0.01328 -0.0118 -0.0079
-0.0011 -0.00179 -0.01328 0.0265 0.0098 0.0045
0.00055 0.00026 -0.0118 0.0098 0.02616 -0.0001
0.0033 0.0028 -0.00792 0.0045 -0.00012 0.0297];
%=============== Generator limits in per unit scale========================
PTmin=[0.05 0.05 0.05 0.05 0.05 0.05];
PTmax=[0.5 0.6 1.0 1.2 1.0 0.6];
%=========================================================================
%%================Initial values of Lagrange's algorithm===================
n=6; % Number of generators
lambda=200; % Initial Lambda
PT=[0.50 0.20 0.2 0.10 0.10 0.12];% Initial Real Powers
m=8000; % Total number of iterations
epsilon=0.01; % Tolerance Value
varepsilon=0.1; % Tolerance value of deltalambda
%===================Wind power modeling using Weibull ================
windData = [ 9.891 10.785 6.861 14.482 12.184 4.602];% wind speed data
v0 = windData';vbins = 0:0.5:max(windData)+1;
pdwind = fitdist(v0,'weibull');
prated = 7.88*10^6;vin =3; vr =12;vout=25;
pwvbins = zeros(size(vbins));
pwvbins(vbins>=vin & vbins<vr) = prated*(vbins(vbins>=vin &vbins<vr).^2-
vin^2)/(vr^2-vin^2);
pwvbins(vbins>=vr &vbins<=vout)= prated;
pdfv = pdf(pdwind,vbins);
avgpower = trapz(vbins,pdfv.*pwvbins);
pwd = (avgpower/1e6); %Power of each wind power unit
nWindunits = input('Enter number of wind generating units (MW) =');
if isempty(nWindunits),nWindunits = 0;
end
TotalwindPower = (nWindunits*pwd)/100;
```

```
fprintf('Available wind = %.3f MW\n',TotalwindPower);
TotalwindPower = max(Wmin,min(TotalwindPower,Wmax));
% %% COST FUNCTION
% actualpw=TotalwindPower*1e6;
% schedw= 0.8*PD*1e6;
% %cwi=1.75;% direct cost coefficient
% cpw=1;% Penalty cost coefficient for over generation
% crw=3.1;% reserve cost coefficient for under generation
% % dCost = cwi*actualpw;% Direct cost of wind power.
% % This factor exists only if the wind farm is not owned by the grid
% % operator.
% UE = cpw*(actualpw-schedw);% Underestimation of wind power
% OE = crw*(schedw-actualpw); % overestimation of wind power
% cost2= (UE + OE)/10^6; % Cost of wind power due to both
% for i=1:nW
% if cost2(i)<=Wmin(i)
% cost2(i)=Wmin(i);
% elseif cost2(i)>=Wmax(i)
% cost2(i)=Wmax(i);
% else cost2(i)=cost2(i);
% end
% end
%================Lagrange's algorithm starts here==================
for iter=1:m
iter
% ===========To solve the WTEED Problem using Lagrange's
algorithm=========================================
fsolve(@WTEED_Casestudy1_funct,PT)
PT=ans;
% P = PT + cost2*10^(-2)
P = PT + TotalwindPower*10^(-2)
% P = max(PTmin,min(P,PTmax));
%=============Checking of generator limits=====================
for i=1:n
if P(i)<=PTmin(i);
P(i)=PTmin(i);
elseif P(i)>=PTmax(i);
P(i)=PTmax(i);
else P(i)=P(i);
end
end
%================== Calculation of Transmission loss=============
PL1=B00;
PL2=zeros(1,n);
for i=1:n
PL2(i)=(B01(i)*P(i));
end
PL2=sum(PL2(:,:));
PL3=zeros(n,n);
for i=1:n
for j=1:n
PL3(i,j)=P(i)*B(i,j)*P(j);
end
end
PL3=sum(PL3(:,:));
PL3=sum(PL3(:,:));
PL=PL1+PL2+PL3;
%=============== Calculation of deltalambda=====================
deltalambda=PD+PL-sum(P(:,:))
if abs(deltalambda)<=epsilon | iter>=m
break
else lambda=lambda+(deltalambda);
```

```
end
end
% ============Lagrange's algorithm ends here========================
%============ Pre-allocation of fuel cost, emission and CEED=========
fuelcost=zeros(1,n);
emissioncost=zeros(1,n);
CEED=zeros(1,n);
%=============== Calculation of fuel cost, emission and CEED===============
for i=1:n
fuelcost(i)=(10.^(-2)*c(i)*P(i)^2+b(i)*P(i)+a(i));
emissioncost(i)=(10.^(-2))*(f(i)*P(i)^2+e(i)*P(i)+d(i));
CEED(i)=(c(i)*P(i)^2+b(i)*P(i)+a(i))+(h(i)*(10.^(-
2))*(f(i)*P(i)^2+e(i)*P(i)+d(i)));
end
%===Printing of Real power, Transmission loss, Fuel cost, Emission and
CEED====
totalfuelcost=sum(fuelcost(:,:))
totalemissioncost=sum(emissioncost(:,:))
CEED=sum(CEED(:,:))
iter
%===============================================Results================
fprintf('\n===========================Final Dispatch
results=============\n');
fprintf('Total Demand PD = %.3f MW\n', PD)
fprintf('Total Wind Supply TotalWindPower = %.3f MW\n', TotalwindPower)
fprintf('Sum of thermal generators = %.3f MW\n', sum(P))
fprintf('Transmission loss PL = %.3f MW\n', PL)
fprintf('Checking Power Balance (PD + PL-(Sumthermal + Pwind))  = %.6f
MW\n',PD+PL-(sum(P)+TotalwindPower));
fprintf('------------------------------------------------------------------
---------------\n');
for i =1:n
  fprintf(' P(%d) = %8.4f(min %.2f,  max %.2f)\n',i,
P(i),PTmin(i),PTmax(i))
end
toc
%%==================================================================
% M-File: WTEED_Casestudy1_funct.m
% IEEE 30 bus system
% ==================M-File Description==============================
% The function is used to calculate the real power using Lagrange's
Algorithm

function F = WTEED_Casestudy1_funct(PT)
global lambda PTmin PTmax n a b c d e f  B B01 h ;
%================ Calculation of price penalty factors (h)===========
h=zeros(1,n); % Pre-allocation of price penalty factor (h)
for i=1:n
h(i)=(c(i)*PTmin(i)^2+b(i)*PTmin(i)+a(i))/((f(i)*PTmax(i)^2+e(i)*PTmax(i)+d
(i)));
end
%============ Calculation of real power of the generators============
for i=1:n
for j=1:n
F(i)=((2*c(i)*PT(i)+b(i))+(2*h(i)*f(i)*PT(i)+(h(i)*e(i))+lambda*(2*PT(j)*B(
i,j)+B01(i))))-lambda;
end
end
end
%==================================================================
```

**APPENDIX A2:** MATLAB script file – WTEED_case10units.m

```
% M-File : WTEED_case10units.m
%IEEE 118 BUS SYSTEM WITH TEN GENERATOR SYSTEM.
%The study focuses on 2000MW power demand as presented in (Mishra and
Shaik, 2024)
% The line data, fuel cost coeffecients and emission co-eficients are also
% take from the above reference.
% ==================M-File Description==============================
% The M-File is used to solve WTEED Problem with using Lagrange’s Algorithm
clear
clc;
tic;
% ==Economic dispatch problem including losses and generator limits==
% ============Initial Lagrange’s variable============================
% lambda=-500; % Initial lambda
propmt='Enter the estimated value of lambda = ';
lambda = input(propmt);
epsil=0.001; % Tolerance value
alfa=0.001; % Incremental deltalambda
n=10; % No of thermal genrators
m=3000; % Total number of iterations
propmt='Enter the Power demand value (in MW)= ';
PD = input(propmt);

% ===========Fuel cost coefficients and Emission Coefficients
=================================
Data=[1000.403   40.5407    0.12951   360.0012   -3.9864    0.04702    10
55;
950.606     39.5804     0.10908   350.0056    -3.9524    0.04652    20     80;
900.705     36.5104     0.12511   330.0056    -3.9023    0.04652    47     120;
800.705     39.5104     0.12111   330.0056    -3.9023    0.04652    20     130;
756.799     38.5390     0.15247   13.8593      0.3277    0.00420    50     160;
451.325     46.1592     0.10587   13.8593      0.3277    0.00420    70     240;
1243.531    38.3055     0.03546   40.2669     -0.5455    0.00680    60     300;
1049.998    40.3965     0.02803   40.2669     -0.5455    0.00680    70     340;
1658.569    36.3278     0.02111   42.8955     -0.5112    0.00460    135    470;
1356.659    38.2704     0.01799   42.8955     -0.5112    0.00460    150    470];
% ===========Fuel cost coefficients=================================
a=Data(:,3);
b=Data(:,2);
c=Data(:,1);
% ======================== Emission Coefficients===================
d=Data(:,6);
e=Data(:,5);
f=Data(:,4);
%=================Generator Limits ===============================
PTmin=Data(:,7);
PTmax=Data(:,8);
nW = 1;
Wmax = 500;
Wmin = 1;
% ==============Transmission loss coefficients======================
B00=0.0055;
B01 = 10^-2.*[0.1 -0.2 -2.8 -0.1 0.1 -0.3 -0.2 -0.2 0.6 3.9]';
B= [0.000049 0.000014 0.000015 0.000015 0.000016 0.000017 0.000017 0.000018
0.000019 0.000020
0.000014 0.000045 0.000016 0.000016 0.000017 0.000015 0.000015 0.000016
0.000018 0.000018
0.000015 0.000016 0.000039 0.000010 0.000012 0.000012 0.000014 0.000014
0.000016 0.000016
```

```
0.000015 0.000016 0.000010 0.000040 0.000014 0.000010 0.000011 0.000012
0.000014 0.000015
0.000016 0.000017 0.000012 0.000014 0.000035 0.000011 0.000013 0.000013
0.000015 0.000016
0.000017 0.000015 0.000012 0.000010 0.000011 0.000036 0.000012 0.000012
0.000014 0.000015
0.000017 0.000015 0.000014 0.000011 0.000013 0.000012 0.000038 0.000016
0.000016 0.000018
0.000018 0.000016 0.000014 0.000012 0.000013 0.000012 0.000016 0.000040
0.000015 0.000016
0.000019 0.000018 0.000016 0.000014 0.000015 0.000014 0.000016 0.000015
0.000042 0.000019
0.000020 0.000018 0.000016 0.000015 0.000016 0.000015 0.000018 0.000016
0.000019 0.000044];
%===============Parameters for wind Power calculations=============
data =[9.891 10.785 6.861 14.482 12.184 4.602 7.232 8.036 12.333 11.074
10.040];
v0= data';
vbins=0:0.5:max(data)+ 1;
%================weibull data distribution================================.
pdwind = fitdist(v0, 'weibull');
prated=5.429*10^6;
vin=5;
vr=15;
vout=45;
%==========================wind velocity constraint=====================
pwvbins = zeros(size(vbins));
powercurve = (vbins>vin)&(vbins<vr);
pwvbins(powercurve) = prated*(vbins(powercurve).^2 - vin^2)./(vr^2-vin^2);
pwvbins(vbins>=vr&vbins<=vout)=prated;
%==========================wind power calculation======================
pdfv=pdf(pdwind,vbins);
avgpower=trapz(vbins,pdfv.*pwvbins);
pwd=(avgpower/(10^6));
nWindunits =input('Enter the number of available wind generators = ');
if isempty(nWindunits)|| nWindunits < 0
nWindunits = 0;
   end
TotalwindPower  =nWindunits*pwd;% Total wind Power from available wind
generators
fprintf('\n\n==============================================================;
fprintf('\n The average wind Power available is = %8.3f
Megawatts\n',TotalwindPower);
fprintf('==============================================================');
if TotalwindPower<Wmin
TotalwindPower=Wmin;
elseif TotalwindPower>Wmax
    TotalwindPower=Wmax;
    end
    % end
    % ==========Calculation of Price Penalty factor=========================
    h=((a.*PTmax.^2+b.*PTmax+c))./((d.*PTmax.^2+e.*PTmax+f));
    %=========================Intial Power and min/max
    limit======================================
    P = min(max((PD-TotalwindPower)/n*ones(n,1),PTmin),PTmax);
    %===============Lagrange's algorithm starts here ===================
    for iter=1:m
    WG = P + (TotalwindPower/n);
    PL = B00 + B01'*WG + WG'*B*WG;
    deltalambda=PD+PL-(sum(P)+ TotalwindPower);
    if mod(iter,500)==0
```

```
fprintf('iter %d:deltalambda = %.6f MW, lambda = %.6f\n',iter,
deltalambda,lambda)
end
if abs(deltalambda)<=epsil
fprintf('\nSolution at iter %d with deltalambda = %.6f
MW\n',iter,deltalambda)
break;
end
% ================Calculation of generator real PTower =========
PT = (lambda - (b+h.*e))./(2.*(a+h.*d));
max(PTmin,min(PTmax,PT));%Generator limit checking
P = PT;
lambda = lambda+alfa*deltalambda;
if lambda <=0
lambda = epsil;
end
if iter == m
warning('Maximum iterations reached without solution. Final deltalambda =
%.6f MW',deltalambda);
end
WG = P + (TotalwindPower/n);
PL = B00 + B01'*WG + WG'*B*WG;
%====================== ================
end
%====== Calculation of fuel cost, emission and CEED values =========
fuelcost=sum(a.*(P.^2)+b.*P+c);
emissioncost= sum(d.*(P.^2)+e.*P+f);
CEED= fuelcost+sum(h.*(d.*(P.^2)+e.*P+f));
%========cost function for wind power====================================
%=====Not considerd in this simulation
actualpw=TotalwindPower*1e6;
schedw= 0.8*PD*1e6;
%cwi=1.75;% cost coefficient
cpw=1;% Penalty cost coefficient for over generation
crw=3.1;% reserve cost coefficient for under generation
%dCost = cwi*actualpw;
UE = cpw*(actualpw-schedw);
OE = crw*(schedw-actualpw);
cost2=(UE + OE)/10^6;
%=====================================================================
%==============================================Results===============
fprintf('\n===========================Final Dispatch
results=============\n');
fprintf('Total Demand PD = %.3f MW\n', PD)
fprintf('Total Wind Supply TotalWindPower = %.3f MW\n', TotalwindPower)
fprintf('Sum of thermal generators = %.3f MW\n', sum(P))
fprintf('Transmission loss PL = %.3f MW\n', PL)
fprintf('Checking Power Balance (PD + PL-(Sumthermal + Pwind))  = %.6f
MW\n',PD+PL-(sum(P)+TotalwindPower));
fprintf('-----------------------------------------------------------------
---------------\n');
for i =1:n
fprintf(' P(%d) = %8.4f(min %.2f,  max %.2f)\n',i, P(i),PTmin(i),PTmax(i))
end
fprintf('-----------------------------------------------------------------
---------------\n');
fprintf('Fuel cost  = %.6f\n',fuelcost);
fprintf('Emission  = %.6f\n',emissioncost);
fprintf('CEED  (fuel+ penalty*emission) = %.6f\n',CEED);
fprintf('Wind rerseve cost = %.6f\n',cost2);
fprintf('Total combined cost   = %.6f\n',fuelcost + cost2);
fprintf('Iterations  = %d\n',iter);
```

```
toc
%%==================================================================
```

**APPENDIX A3:** **MATLAB script file – WTEED_case40units.m**

```
% M-File : WTEED_case40units.m
%IEEE 30 bus with 40 Units GENERATOR SYSTEM with wind power
%The simulation data for power limits, fuel cost coefficients,
%emission coefficients, and B-loss coefficients used in this test system
% are those used in (Secui et al.,2024). This program is used for the table
% 3.10 and 3.11. The wind power can be varied by changing the number of
% available wind turbines and rated wind power.
clearvars; close all; clc;
tic;
lambda = 8.9914;      % initial Lagrange multiplier
epsilon = 1e-3;       % convergence tolerance (MW)
alfa = 1e-3;          % lambda update step size
n = 40;               % number of thermal generators
m = 35000;            % max iterations
PD = input('Enter the Power demand value (in MW) = ');
% ===========Fuel cost coefficients, Emission Coefficients, and limits of a
generator =================================
data1 =    [36      114    0.0069 6.73    94.705     0.048  -2.22  60
      36      114    0.0069 6.73    94.705 0.048  -2.22  60
      60      120    0.02028      7.07    309.54 0.0762 -2.36  100
      80      190    0.00942      8.18    369.03 0.054  -3.14  120
      47      97     0.0114 5.35    148.89 0.085  -1.89  50
      68      140    0.01142      8.05    222.33 0.0854 -3.08  80
      110     300    0.00357      8.03    287.71 0.0242 -3.06  100
      135     300    0.00492      6.99    391.98 0.031  -2.32  130
      135     300    0.00573      6.6        455.76    0.0335 -2.11  150
      130     300    0.00605      12.9    722.82 0.425  -4.34  280
       94     375    0.00515      12.9    635.2  0.0322 -4.34  220
       94     375    0.00569      12.8    654.69 0.0338 -4.28  225
      125     500    0.00421      12.5    913.4  0.0296 -4.18  300
      125     500    0.00752      8.84    1760.4 0.0512 -3.34  520
      125     500    0.00708      9.15    1728.3 0.0496 -3.55  510
      125     500    0.00708      9.15    1728.3 0.0496 -3.55  510
      220     500    0.00313      7.97    647.85 0.0151 -2.68  220
      220     500    0.00313      7.95    649.69 0.0151 -2.66  222
      242     550    0.00313      7.97    647.83 0.0151 -2.68  220
      242     550    0.00313      7.97    647.81 0.0151 -2.68  220];

data2 =   [254      550    0.00298      6.63    785.96 0.0145 -2.22  290
          254      550    0.00298      6.63    785.96 0.0145 -2.22  285
           254     550    0.00284      6.66    794.53 0.0138 -2.26  295
           254     550    0.00284      6.66    794.53 0.0138 -2.26  295

           254     550    0.00277      7.1        801.32    0.0132 -2.42  310
            254    550    0.00277      7.1        801.32    0.0132 -2.42  310

            10     150    0.52124      3.33    1055.1 1.842  -1.11  360
             10    150    0.52124      3.33    1055.1 1.842  -1.11  360
             10    150    0.52124      3.33    1055.1 1.842  -1.11  360
            47     97     0.0114 5.35    148.89 0.085  -1.89  50
              60   190    0.0016 6.43    222.92 0.0121 -2.08  80
              60   190    0.0016 6.43    222.92 0.0121 -2.08  80
              60   190    0.0016 6.43    222.92 0.0121 -2.08  80
              90   200    0.0001 8.95    107.87 0.0012 -3.48  65
              90   200    0.0001 8.62    116.58 0.0012 -3.24  70
               90  200    0.0001 8.62    116.58 0.0012 -3.24  70
              25   110    0.0161 5.88    307.45 0.095  -1.98  100
              25   110    0.0161 5.88    307.45 0.095  -1.98  100
```

```
            25      110     0.0161 5.88     307.45 0.095   -1.98  100
      242   550     0.00313        7.97      647.83      0.0151 -2.68  220];

data = [data1;data2];

%==================Generator Limits ===============================
Pmin=data(:,1);
Pmax=data(:,2);
Wmax = 2000;
Wmin = 1;
% ==========Fuel cost coefficients=================================
a=data(:,3);
b=data(:,4);
c=data(:,5);
% ======================= Emission Coefficients===================
d=data(:,6);
e=data(:,7);
f=data(:,8);
% =============Transmission loss coefficients=====================
B00=0.56;
B01=10^-4.*[-3.908  -1.297 7.047  0.591  2.161  -6.635 -3.908 -1.297 7.047...

            0.591   2.161  -6.635 -3.908 -1.297 7.047  0.591  2.161  -6.635...

            -3.908  -1.297 7.047  0.591  2.161  -6.635 -3.908 -1.297 7.047...

            0.591   2.161  -6.635 -3.908 -1.297 7.047  0.591  2.161  -6.635...
            -3.908  -1.297 7.047  0.591]';

B1_half=[17  12     7      -1     -5     -2     17     12     7      -1     -5
-2     17     12     7      -1     -5     -2     17     12
  12   14     9      1      -6     -1     12     14     9      1      -6     -1
12     14     9      1      -6     -1     12     14
  7    9      31     0      -10    -6     7      9      31     0      -10    -6
7      9      31     0      -10    -6     7      9
  -1   1      0      24     -6     -8     -1     1      0      24     -6     -8
-1     1      0      24     -6     -8     -1     1
  -5 -6  -10  -6     129    -2     -5     -6     -10    -6     129    -2     -5
-6     -10    -6     129    -2     -5     -6
  -2   -1     -6     -8     -2     150    -2     -1     -6     -8     -2     150
-2     -1     -6     -8     -2     150    -2     -1
  17   12     7      -1     -5     -2     17     12     7      -1     -5     -2
17     12     7      -1     -5     -2     17     12
  12   14     9      1      -6     -1     12     14     9      1      -6     -1
12     14     9      1      -6     -1     12     14
  7    9      31     0      -10    -6     7      9      31     0      -10    -6
7      9      31     0      -10    -6     7      9
 -1    1      0      24     -6     -8     -1     1      0      24     -6     -8
-1     1      0      24     -6     -8     -1     1
 -5   -6-10   -6     129    -2     -5     -6     -10    -6     129    -2     -5
-6     -10    -6     129    -2     -5     -6
 -2    -1     -6     -8     -2     150    -2     -1     -6     -8     -2     150
-2     -1     -6     -8     -2     150    -2     -1
  17   12     7      -1     -5     -2     17     12     7      -1     -5     -2
17     12     7      -1     -5     -2     17     12
  12   14     9       1     -6     -1     12     14     9      1      -6     -1
12     14     9      1      -6     -1     12     14
  7    9      31     0      -10  -6 7    9      31     0      -10    -6     7
9      31     0      -10    -6     7      9
  -1   1      0      24     -6     -8     -1     1      0      24     -6     -8
-1     1      0      24     -6     -8     -1     1
```

```
 -5    -6     -10  -6 129    -2     -5     -6     -10    -6     129    -2     -5
-6     -10    -6     129    -2     -5     -6
 -2    -1     -6     -8     -2     150    -2     -1     -6     -8     -2     150
-2     -1     -6     -8     -2     150    -2     -1
 17    12     7      -1     -5     -2     17     12     7      -1     -5     -2
17     12     7      -1     -5     -2     17     12
 12    14     9      1      -6     -1     12     14     9      1      -6     -1
12     14     9      1      -6     -1     12     14];

B2_half=[7    -1     -5     -2     17     12     7      -1     -5     -2     17
12     7      -1     -5     -2     17     12     7      -1
  9    1      -6     -1     12     14     9      1      -6     -1     12     14
9      1      -6     -1     12     14     9      1
  31   0      -10    -6     7      9      31     0      -10    -6     7      9
31     0      -10    -6     7      9      31     0
  0    24     -6     -8     -1     1      0      24     -6     -8     1      1
0      24     -6     -8     -1     1      0      24
 -10   -6     129    -2     -5     -6     -10    -6     129    -2     -5     -6
-10    -6     129    -2     -5     -6     -10    -6
 -6    -8     -2     150    -2     -1     -6     -8     -2     150    -2     -1
-6     -8     -2     150    -2     -1     -6     -8
  7    -1     -5     -2     17     12     7      -1     -5     -2     17     12
7      -1     -5     -2     17     12     7      -1
  9    1      -6     -1     12     14     9      1      -6     -1     12     14
9      1      -6     -1     12     14     9      1
  31   0      -10    -6     7      9      31     0      -10    -6     7      9
31     0      -10    -6     7      9      31     0
  0    24     -6     -8     -1     1      0      24     -6     -8     -1     1
0      24     -6     -8     -1     1      0      24
 -10   -6     129    -2     -5     -6     -10    -6     129    -2     -5     -6
-10    -6     129    -2     -5     -6     -10    -6
 -6    -8     -2     150    -2     -1     -6     -8     -2     150    -2     -1
-6     -8     -2     150    -2     -1     -6     -8
  7    -1     -5     -2     17     10     -6     7      9      31     0      -10
-6     7      9      31     0      -10    -6     7
 0     24     -6     -8     -1     1      0      24     -6     -8     -1     1
0      24     -6     -8     -1     1       0     24
 -10   -6     129    -2     -5     -6     -10    -6     129    -2     -5     -6
-10    -6     129    -2     -5     -6     -10    -6
 -6    -8     12     7      -1     -5     -2     17     12     7      -1     -5
-2     17     12     7      -1   9    31   0
 9     1      -6     -1     12     14     9      1      -6     -1     12     14
9      1      -6     -1     12     14      9      1
 31    0      -2     150    -2     -1     -6     -8     -2     150    -2     -1
-6     -8     -2     150    -2     -1     -6     -8
 7     -1     -5     -2     17     12     7      -1     -5     -2     17     12
7      -1     -5     -2     17     12     7      -1
 9     1      -6     -1     12     14     9      1      -6     -1     12     14
9      1      -6     -1     12     14     9      1];

B3_half = [  7      9      31     0      -10    -6     7      9      31     0
-10    -6     7      9      31     0      -10    -6     7      9
      -1      1      0      24     -6     -8     -1     1      0      24     -6
-8     -1     1      0      24     -6     -8     -1     1
      -5      -6     -10    -6     129    -2     -5     -6     -10    -6     129
-2     -5     -6     -10    -6     129    -2     -5     -6
      -2      -1     -6     -8     -2     150    -2     -1     -6     -8     -2
150    -2     -1     -6     -8     -2     150    -2     -1
      17      12     7      -1     -5     -2     17     12     7      -1     -5
-2     17     12     7      -1     -5     -2     17     12
      12      14     9      1      -6     -1     12     14     9      1      -6
-1     12     14     9      1      -6     -1     12     14
```

```
	79	31	0	-10	-6	7	9	31	0	-10	-6
7	9	31	0	-10	-6	7	9
	-1	1	0	24	-6	-8	-1	1	0	24	-6
-8	-1	1	0	24	-6	-8	-1	1
	-5	-6	-10	-6	129	-2	-5	-6	-10	-6	129
-2	-5	-6	-10	-6	129	-2	-5	-6
	-2	-1	-6	-8	-2	150	-2	-1	-6	-8	-2
150	-2	-1	-6	-8	-2	150	-2	-1
	17	12	7	-1	-5	-2	17	12	7	-1	-5
-2	17	12	7	-1	-5	-2	17	12
	12	14	9	1	-6	-1	12	14	9	1	-6
-1	12	14	9	1	-6	-1	12	14
	79	31	0	-10	-6	7	9	31	0	-10	-6
7	9	31	0	-10	-6	7	9
	-1	1	0	24	-6	-8	-1	1	0	24	-6
-8	-1	1	0	24	-6	-8	-1	1
	-5	-6	-10	-6	129	-2	-5	-6	-10	-6	129
-2	-5	-6	-10	-6	129	-2	-5	-6
	-2	-1	-6	-8	-2	150	-2	-1	-6	-8	-2
150	-2	-1	-6	-8	-2	150	-2	-1
	17	12	7	-1	-5	-2	17	12	7	-1	-5
-2	17	12	7	-1	-5	-2	17	12
	12	14	9	1	-6	-1	12	14	9	1	-6
-1	12	14	9	1	-6	-1	12	14
	79	31	0	-10	-6	7	9	31	0	-10	-6
7	9	31	0	-10	-6	7	9
	-1	1	0	24	-6	-8	-1	1	0	24	-6
-8	-1	1	0	24	-6	-8	-1	1];

B4_half =[	31	0	-10	-6	7	9	31	0	-10	-6
7	9	31	0	-10	-6	7	9	31	0
	024	-6	-8	-1	1	0	24	-6	-8	-1	1
0	24	-6	-8	-1	1	0	24
	-10	-6	129	-2	-5	-6	-10	-6	129	-2	-5
-6	-10	-6	129	-2	-5	-6	-10	-6
	-6	-8	-2	150	-2	-1	-6	-8	-2	150	-2
-1	-6	-8	-2	150	-2	-1	-6	-8
	7-1	-5	-2	17	12	7	-1	-5	-2	17	12
7	-1	-5	-2	17	12	7	-1
	91	-6	-1	12	14	9	1	-6	-1	12	14
9	1	-6	-1	12	14	9	1
	31	0	-10	-6	7	9	31	0	-10	-6	7
9	31	0	-10	-6	7	9	31	0
	024	-6	-8	-1	1	0	24	-6	-8	-1	1
0	24	-6	-8	-1	1	0	24
	-10	-6	129	-2	-5	-6	-10	-6	129	-2	-5
-6	-10	-6	129	-2	-5	-6	-10	-6
	-6	-8	-2	150	-2	-1	-6	-8	-2	150	-2
-1	-6	-8	-2	150	-2	-1	-6	-8
	7-1	-5	-2	17	12	7	-1	-5	-2	17	12
7	-1	-5	-2	17	12	7	-1
	91	-6	-1	12	14	9	1	-6	-1	12	14
9	1	-6	-1	12	14	9	1
	31	0	-10	-6	7	9	31	0	-10	-6	7
9	31	0	-10	-6	7	9	31	0
	024	-6	-8	-1	1	0	24	-6	-8	-1	1
0	24	-6	-8	-1	1	0	24
	-10	-6	129	-2	-5	-6	-10	-6	129	-2	-5
-6	-10	-6	129	-2	-5	-6	-10	-6
	-6	-8	-2	150	-2	-1	-6	-8	-2	150	-2
-1	-6	-8	-2	150	-2	-1	-6	-8
```

```
     7-1     -5     -2     17     12     7      -1     -5     -2     17     12
7     -1     -5     -2     17     12     7      -1
     91      -6     -1     12     14     9      1      -6     -1     12     14
9     1      -6     -1     12     14     9      1
     31      0      -10    -6     7      9      31     0      -10    -6     7
9     31     0      -10    -6     7      9      31     0
     024     -6     -8     -1     1      0      24     -6     -8     -1     1
0     24     -6     -8     -1     1      0      24];

B= 10^-6.*[B1_half B2_half;B3_half B4_half ];
%================Parameters for wind Power calculations==============
data =[9.891 10.785 6.861 14.482 12.184 4.602 7.232 8.036 12.333 11.074
10.040];
v0= data';
vbins=0:0.5:max(data)+ 1;
%=================weibull data
distribution=================================.
pdwind = fitdist(v0, 'weibull');
prated=5.429*10^6;
vin=5;
vr=15;
vout=45;
%===========================wind velocity constraint=====================
pwvbins = zeros(size(vbins));
powercurve = (vbins>vin)&(vbins<vr);
pwvbins(powercurve) = prated*(vbins(powercurve).^2 - vin^2)./(vr^2-vin^2);
pwvbins(vbins>=vr&vbins<=vout)=prated;
%============================wind power calculation=======================
pdfv=pdf(pdwind,vbins);
avgpower=trapz(vbins,pdfv.*pwvbins);
pwd=(avgpower/(10^6));
nWindunits =input('Enter the number of available wind generators = ');
if isempty(nWindunits)|| nWindunits < 0
nWindunits = 0;
end
TotalwindPower  =nWindunits*pwd;% Total wind Power from available wind
generators
fprintf('\n\n============================================================='
);
fprintf('\n The average wind Power available is = %8.3f
Megawatts\n',TotalwindPower);
fprintf('============================================================');
if TotalwindPower<Wmin
TotalwindPower=Wmin;
elseif TotalwindPower>Wmax
TotalwindPower=Wmax;
end
% end
% ===========Calculation of Price Penalty factor==========================
h=((a.*Pmax.^2+b.*Pmax+c))./((d.*Pmax.^2+e.*Pmax+f));
%==========================Intial Power and min/max
limit=======================================
P = min(max((PD-TotalwindPower)/n*ones(n,1),Pmin),Pmax);
%================Lagrange's algorithm starts here ===================
for iter = 1:m
G = P + (TotalwindPower / n);
PL = B00 + B01' * G + G' * B * G;
deltalambda = PD + PL - (sum(P) + TotalwindPower);
if abs(deltalambda) <= epsilon
fprintf('\nConverged at iter %d: deltalambda = %.6f MW\n', iter,
deltalambda);
break;
```

```
    end
    PT = (lambda - (b + h.*e)) ./ (2 .* (a + h.*d));
    PT = max(Pmin, min(Pmax, PT));
    P = PT;
    lambda = max(eps, lambda + alfa * deltalambda);
    if mod(iter, 500) == 0
        fprintf('iter %d: mismatch = %.6f, lambda = %.6f\n', iter,
deltalambda, lambda);
    end
end
if iter == m
warning('Max iterations reached. Final mismatch = %.6f MW', deltalambda);
end
%===========================================================================
%====== Calculation of fuel cost, emission and CEED values ================
G = P + (TotalwindPower / n);
PL = B00 + B01' * G + G' * B * G;
fuelCost = sum(a .* P.^2 + b .* P + c);
emissionCost = sum(d .* P.^2 + e .* P + f);
CEED = fuelCost + sum(h .* (d .* P.^2 + e .* P + f));
%===========================================================================
%========cost function for wind power======================================
%=====Not considerd in this simulation
UE = 1*(TotalwindPower*1e6 - 0.8*PD*1e6);
OE = 3.1*(0.8*PD*1e6 - TotalwindPower*1e6);
windCost = (UE + OE) / 1e6;
%==========================================================================
%===========================================================================
%================================================Results================
fprintf('\n=== Results ===\n');
fprintf('PD = %.3f MW, Wind supply = %.3f MW, Thermal sum = %.3f MW\n', PD,
TotalwindPower, sum(P));
fprintf('Loss PL = %.6f MW\nBalance check: PD + PL - (th + wind) = %.6f
MW\n', PL, PD + PL - (sum(P) + TotalwindPower));
disp('--- Unit outputs ---');
for i = 1:n
fprintf('P(%2d) = %.4f MW (min %.2f, max %.2f)\n', i, P(i), Pmin(i),
Pmax(i));
end
fprintf('Fuel cost = %.6f\nEmission cost = %.6f\nCEED = %.6f\nWind cost
term = %.6f\nTotal cost ≈ %.6f\n', ...
fuelCost, emissionCost, CEED, windCost, fuelCost + windCost);
fprintf('Iterations = %d\n', iter);
toc;
```

**APPENDIX B: DEVELOPED MATLAB PROGRAM FOR A SINGLE AREA WTEED PROBLEM USING PSO ALGORITHM**

**APPENDIX B1: MATLAB script file – WTEED_casePSO6units.m**

```
%This is IEEE30 bus system with 6 generathing units. Named:
WTEED_casePSO6units.m.
% The program is created using a per unit scale. But table 4.1 is recreated
% back to normal scale.This program is used to create Table 4.1, and is
%also used in table 4.2. It is using a function called objective2.m
%
clearvars;clc; close all;tic;
%==========User to enter power demand of the system=======
PD = input('What is the Power demand in (MW)=');
n =6; %number of used generators
num_particles = 50;% number of particles in the population
maxIter = 1000; %Number of iterations
w = 0.9; c1 = 1.5; c2 = 1.5;
penaltyEmissionFactor = 10000;
powerBalancePenalty = 1000000;
printInterval = 10;
nW = 1;
Wmax = 500;
Wmin = 1;
%============Fuel cost with valve point effect coefficients=========
a=[10 10 20 10 20 10]';
b=[200 150 180 100 180 150]';
c=[100 120 40 60 40 100]';
%% Emission with valve point effect coefficients
d=[4.091 2.543 4.258 5.326 4.258 6.131]';
e=[-5.554 -6.047 -5.094 -3.55 -5.094 -5.555]';
f=[6.49 5.638 4.586 3.38 4.586 5.151]';
Wmin = 1;
Wmax = 2000;
nW = 1;
Pmin=[0.05 0.05 0.05 0.05 0.05 0.05]';
Pmax=[0.5 0.6 1.0 1.2 1.0 0.6]';
% ==============Transmission loss coefficients======================
B00=0.0014;
B01=[0.010731; 1.7704; -4.0645; 3.8453; 1.3832; 5.5503]*10.^(-3);
B=[ 0.0218  0.0107  -0.00036 -0.0011  0.00055  0.0033
    0.0107  0.01704 -0.0001  -0.00179 0.00026  0.0028
   -0.0004 -0.0002   0.02459 -0.01328 -0.0118 -0.0079
   -0.0011 -0.00179 -0.01328  0.0265   0.0098   0.0045
    0.00055 0.00026 -0.0118   0.0098   0.02616 -0.0001
    0.0033  0.0028  -0.00792  0.0045 -0.00012  0.0297];
%================Parameters for wind Power calculations=============
data =[9.891 10.785 6.861 14.482 12.184 4.602 7.232 8.036 12.333 11.074
10.040];
v0= data';
vbins=0:0.5:max(data)+ 1;
%=================weibull data
distribution================================.
pdwind = fitdist(v0, 'weibull');
prated=11.429*10^6;
vin=5;
vr=15;
vout=45;
%===========================wind velocity constraint=====================
pwvbins = zeros(size(vbins));
powercurve = (vbins>vin)&(vbins<vr);
pwvbins(powercurve) = prated*(vbins(powercurve).^2 - vin^2)./(vr^2-vin^2);
```

```
pwvbins(vbins>=vr&vbins<=vout)=prated;
%============================wind power calculation======================
pdfv=pdf(pdwind,vbins);
avgpower=trapz(vbins,pdfv.*pwvbins);
pwd=(avgpower/(10^6));
nWindunits =input('Enter the number of available wind generators = ');
if isempty(nWindunits)|| nWindunits < 0
    nWindunits = 0;
end
TotalwindPower  =nWindunits*pwd/100;% Total wind Power from available wind
generators
fprintf('\n\n============================================================='
);
fprintf('\n The average wind Power available is = %8.3f
Megawatts\n',TotalwindPower);
fprintf('=============================================================');
TotalwindPower = max(Wmin,min(TotalwindPower,Wmax));
%==================================
%%================= Initial variables of the PSO algorithm=================
dim = n; lb = Pmin; ub = Pmax;
vmax = 0.2*(ub-lb); vmin = -vmax;
X =zeros(num_particles,dim); V = zeros(num_particles,dim);
pbest = X; pbestVal =inf(num_particles,1); gbest = zeros(1,dim);gbestVal
=inf;
 for i=1:num_particles
    X(i,:)=(lb+(ub-lb).*rand(dim,1))';
    V(i,:)= (vmin+(vmax-vmin).*rand(dim,1))';
[val,~]=objective2(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyE
missionFactor,powerBalancePenalty);
    pbest(i,:)=X(i,:);
    pbestVal(i)=val;
    if val<gbestVal
        gbest=X(i,:);
        gbestVal = val;
    end
 end
% ==================Main loop of the PSO algorithm=====================
for iter =1:maxIter
for i=1:num_particles
        r1 =rand(1,dim);r2 =rand(1,dim);
        V(i,:)=w*V(i,:)+c1*r1.*(pbest(i,:)-X(i,:))+c2*r2.*(gbest-X(i,:));
        V(i,:)=max(min(V(i,:),vmax'),vmin');
        X(i,:)=X(i,:)+V(i,:);
        X(i,:)=max(min(X(i,:),ub'),lb')
[val,~]=objective2(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyE
missionFactor,powerBalancePenalty);
        if val<pbestVal(i), pbestVal(i)= val; pbest(i,:)= X(i,:);end
        if val<gbestVal,gbestVal = val; gbest = X(i,:);
        end
     if mod(iter,printInterval)==0
        fprintf('Iter %d CEED=%.3f\n',iter,gbestVal);
    end
end
end
%==============================Final evaluation of
variables==============================
[Pcost,details]=objective2(gbest',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,p
enaltyEmissionFactor,powerBalancePenalty);
fprintf('\n==========Final PSO Dispatch (6units)===========\n');
for i=1:n
    fprintf(' P(%d)=%.3f MW (min %.1f max
%.1f)\n',i,gbest(i),Pmin(i),Pmax(i))
```

```
end
fprintf('Sum Thermal=%.3f MW, Wind=%.3f MW, Loss=%.3f
MW\n',sum(gbest),TotalwindPower,details.PL)
      fprintf('Fuel Cost=%.3f , Emission=%.3f ,
CEED=%.3f\n',details.fuelcost,details.emission,Pcost);
toc;
% This function is called on the main program: WTEED_casePSO6units for
calculation
% of PL,fuelcost,emission.
function
[cost,details]=objective2(P,PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,k1,kpb)
    P =P(:); WG = P+(TotalwindPower/length(P));
    PL = B00+B01'*WG+WG'*B*WG; % Calculation of Power losses
    fuel=sum(a.*(P.^2)+b.*P+c); %calculation of fuel cost function
    em =sum((10.^(-2).*(d.*(P.^2)+e.*P+f)));% Emission calculation
    % em =sum((d.*(P.^2)+e.*P+f));
    CEED = fuel+k1*em;% Combined emission calculation
    % CEED = fuel+k1*em.*(10.^(-2));
    mismatch=PD+PL-(sum(P)+TotalwindPower); % Total power balance
    penalty=kpb*mismatch^2;
    cost =CEED+penalty;
    details.PL=PL;details.fuelcost=fuel;details.emission=em;% all values
that are defined
    % under details to be called on the main script file.
    en
```

**APPENDIX B2:** **MATLAB script file – WTEED_casePSO10units.m**

```
%%%======================Main program===================================
%This is IEEE30 bus system with 10 generathing units. Named:
WTEED_casePSO10units.m.
% The program is used to create table 4.3 and also used in Table 4.4
%  It is using a function called objective.m
clearvars;clc; close all;tic;
%==========User to enter power demand of the system=======
PD = input('What is the Power demand in (MW)=');
n =10; %number of used generators
num_particles = 40;% number of particles in the population
maxIter = 500; %Number of iterations
w = 0.7; c1 = 1.5; c2 = 1.5;
penaltyEmissionFactor = 1;
powerBalancePenalty = 1000000;
printInterval = 10;

% ===========Fuel cost coefficients and Emission Coefficients
=================================
Data=[1000.403   40.5407    0.12951   360.0012   -3.9864    0.04702    10
55;
      950.606    39.5804    0.10908   350.0056   -3.9524    0.04652    20
80;
      900.705    36.5104    0.12511   330.0056   -3.9023    0.04652    47
120;
      800.705    39.5104    0.12111   330.0056   -3.9023    0.04652    20
130;
      756.799    38.5390    0.15247   13.8593     0.3277    0.00420    50
160;
      451.325    46.1592    0.10587   13.8593     0.3277    0.00420    70
240;
      1243.531   38.3055    0.03546   40.2669    -0.5455    0.00680    60
300;
      1049.998   40.3965    0.02803   40.2669    -0.5455    0.00680    70
340;
```

```
      1658.569   36.3278    0.02111   42.8955    -0.5112    0.00460    135
470;
      1356.659   38.2704    0.01799   42.8955    -0.5112    0.00460    150
470];
% ===========Fuel cost coefficients=================================
a=Data(:,3);
b=Data(:,2);
c=Data(:,1);
% ======================== Emission Coefficients====================
d=Data(:,6);
e=Data(:,5);
f=Data(:,4);
%==================Generator Limits ===============================
Pmin=Data(:,7);
Pmax=Data(:,8);
nW = 1;
Wmax = 500;
Wmin = 1;
% ==============Transmission loss coefficients======================
B00=0.0055;
B01=10^-2.*[ 0.1 -0.2 -2.8 -0.1 0.1 -0.3 -0.2 -0.2 0.6 3.9]';
B= [ 0.000049 0.000014 0.000015 0.000015 0.000016 0.000017 0.000017
0.000018 0.000019 0.000020
    0.000014 0.000045 0.000016 0.000016 0.000017 0.000015 0.000015 0.000016
0.000018 0.000018
    0.000015 0.000016 0.000039 0.000010 0.000012 0.000012 0.000014 0.000014
0.000016 0.000016
    0.000015 0.000016 0.000010 0.000040 0.000014 0.000010 0.000011 0.000012
0.000014 0.000015
    0.000016 0.000017 0.000012 0.000014 0.000035 0.000011 0.000013 0.000013
0.000015 0.000016
    0.000017 0.000015 0.000012 0.000010 0.000011 0.000036 0.000012 0.000012
0.000014 0.000015
    0.000017 0.000015 0.000014 0.000011 0.000013 0.000012 0.000038 0.000016
0.000016 0.000018
    0.000018 0.000016 0.000014 0.000012 0.000013 0.000012 0.000016 0.000040
0.000015 0.000016
    0.000019 0.000018 0.000016 0.000014 0.000015 0.000014 0.000016 0.000015
0.000042 0.000019
    0.000020 0.000018 0.000016 0.000015 0.000016 0.000015 0.000018 0.000016
0.000019 0.000044];
%================Parameters for wind Power calculations=============
data =[9.891 10.785 6.861 14.482 12.184 4.602 7.232 8.036 12.333 11.074
10.040];
v0= data';
vbins=0:0.5:max(data)+ 1;
%=================weibull data
distribution=================================.
pdwind = fitdist(v0, 'weibull');
prated=11.429*10^6;
vin=5;
vr=15;
vout=45;
%===========================wind velocity constraint=====================
pwvbins = zeros(size(vbins));
powercurve = (vbins>vin)&(vbins<vr);
pwvbins(powercurve) = prated*(vbins(powercurve).^2 - vin^2)./(vr^2-vin^2);
pwvbins(vbins>=vr&vbins<=vout)=prated;
%============================wind power calculation======================
pdfv=pdf(pdwind,vbins);
avgpower=trapz(vbins,pdfv.*pwvbins);
pwd=(avgpower/(10^6));
```

```
nWindunits =input('Enter the number of available wind generators = ');
if isempty(nWindunits)|| nWindunits < 0
    nWindunits = 0;
end
TotalwindPower  =nWindunits*pwd;% Total wind Power from available wind
generators
fprintf('\n\n=============================================================='
);
fprintf('\n The average wind Power available is = %8.3f
Megawatts\n',TotalwindPower);
fprintf('==============================================================');
TotalwindPower = max(Wmin,min(TotalwindPower,Wmax));
%% Initial variables of the PSO algorithm
dim = n; lb = Pmin; ub = Pmax;
vmax = 0.5*(ub-lb); vmin = -vmax;
X =zeros(num_particles,dim); V = zeros(num_particles,dim);
pbest = X; pbestVal =inf(num_particles,1); gbest = zeros(1,dim);gbestVal
=inf;
 for i=1:num_particles
    X(i,:)=(lb+(ub-lb).*rand(dim,1))';
    V(i,:)= (vmin+(vmax-vmin).*rand(dim,1))';

[val,~]=objective(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyEm
issionFactor,powerBalancePenalty);
    pbest(i,:)=X(i,:);
    pbestVal(i)=val;
    if val<gbestVal
        gbest=X(i,:);
        gbestVal = val;
    end
 end
% ==================Main loop of the PSO algorithm======================
for iter =1:maxIter
for i=1:num_particles
        r1 =rand(1,dim);r2 =rand(1,dim);
        V(i,:)=w*V(i,:)+c1*r1.*(pbest(i,:)-X(i,:))+c2*r2.*(gbest-X(i,:));
        V(i,:)=max(min(V(i,:),vmax'),vmin');
        X(i,:)=X(i,:)+V(i,:);
        X(i,:)=max(min(X(i,:),ub'),lb');

[val,~]=objective(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyEm
issionFactor,powerBalancePenalty);
        if val<pbestVal(i), pbestVal(i)= val; pbest(i,:)= X(i,:);end
        if val<gbestVal,gbestVal = val; gbest = X(i,:);
        end
     if mod(iter,printInterval)==0
        fprintf('Iter %d CEED=%.3f\n',iter,gbestVal);
    end
end
end
%==============================Final evaluation of
variables==============================
[Pcost,details]=objective(gbest',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,pe
naltyEmissionFactor,powerBalancePenalty);

fprintf('\n==========Final PSO Dispatch (6units)===========\n');
for i=1:n
    fprintf(' P(%d)=%.3f MW (min %.1f max
%.1f)\n,',i,gbest(i),Pmin(i),Pmax(i))
end
fprintf('Sum Thermal=%.3f MW, Wind=%.3f MW, Loss=%.3f
MW\n',sum(gbest),TotalwindPower,details.PL)
```

```
fprintf('Fuel Cost=%.3f , Emission=%.3f ,
CEED=%.3f\n',details.fuelcost,details.emission,Pcost);
toc;

% This function is called on the main program: WTEED_casePSO10units for
calculation
% of PL,fuelcost,emission.Named: objective.m
function[cost,details]=objective(P,PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,
k1,kpb)
    P =P(:); WG = P+(TotalwindPower/length(P));
    PL = B00+B01'*WG+WG'*B*WG;
    fuel=sum(a.*(P.^2)+b.*P+c);
    em =sum(d.*(P.^2)+e.*P+f);
    CEED = fuel+k1*em;
    mismatch=PD+PL-(sum(P)+TotalwindPower);
    penalty=kpb*mismatch^2;
    cost =CEED+penalty;
    details.PL=PL;details.fuelcost=fuel;details.emission=em;
end
%==========================================================================
=
```

**APPENDIX B3:** **MATLAB script file – WTEED_casePSO40units.m**

```
%%%======================Main program===================================
%This is IEEE30 bus system with 40 generathing units. Named:
WTEED_casePSO40units.m.
% The program is used to create table 4.5 and also used in Table 4.6
%  It is using a function called objective.m
% This is IEEE30 bus system with 40 generathing units.
clear all; clc; close all;tic;
%==========User to enter power demand of the system=======
PD = input('What is the Power demand in (MW)=');
n =40; %number of used generators
num_particles = 40;% number of particles in the population
maxIter = 1000; %Number of iterations
w = 0.65; c1 = 1.5; c2 = 1.5;
penaltyEmissionFactor = 10000;
powerBalancePenalty = 1000000;
printInterval = 10;
% ===========Fuel cost coefficients, Emission Coefficients, and limits of a
generator ==================================
data1 =     [36      114     0.0069 6.73     94.705      0.048  -2.22  60
             36      114     0.0069 6.73    94.705 0.048  -2.22  60
             60      120     0.02028        7.07    309.54 0.0762 -2.36  100
             80      190     0.00942        8.18    369.03 0.054  -3.14  120
             47      97      0.0114 5.35    148.89 0.085  -1.89  50
             68      140     0.01142        8.05    222.33 0.0854 -3.08  80
             110     300     0.00357        8.03    287.71 0.0242 -3.06  100
             135     300     0.00492        6.99    391.98 0.031  -2.32  130
             135     300     0.00573        6.6         455.76     0.0335 -2.11  150
             130     300     0.00605        12.9    722.82 0.425  -4.34  280
              94     375     0.00515        12.9    635.2  0.0322 -4.34  220
              94     375     0.00569        12.8    654.69 0.0338 -4.28  225
             125     500     0.00421        12.5    913.4  0.0296 -4.18  300
             125     500     0.00752        8.84    1760.4 0.0512 -3.34  520
             125     500     0.00708        9.15    1728.3 0.0496 -3.55  510
             125     500     0.00708        9.15    1728.3 0.0496 -3.55  510
             220     500     0.00313        7.97    647.85 0.0151 -2.68  220
             220     500     0.00313        7.95    649.69 0.0151 -2.66  222
             242     550     0.00313        7.97    647.83 0.0151 -2.68  220
             242     550     0.00313        7.97    647.81 0.0151 -2.68  220];
```

```
data2 =    [254      550    0.00298       6.63    785.96 0.0145 -2.22  290
      254     550    0.00298       6.63    785.96 0.0145 -2.22  285
         254   550    0.00284       6.66    794.53 0.0138 -2.26  295
         254   550    0.00284       6.66    794.53 0.0138 -2.26  295

         254   550    0.00277       7.1        801.32    0.0132 -2.42  310
      254     550    0.00277       7.1        801.32    0.0132 -2.42  310
      10      150    0.52124       3.33    1055.1 1.842  -1.11  360
         10   150    0.52124       3.33    1055.1 1.842  -1.11  360
         10   150    0.52124       3.33    1055.1 1.842  -1.11  360
       47      97     0.0114 5.35    148.89 0.085  -1.89  50
       60      190    0.0016 6.43    222.92 0.0121 -2.08  80
       60      190    0.0016 6.43    222.92 0.0121 -2.08  80
       60      190    0.0016 6.43    222.92 0.0121 -2.08  80
       90      200    0.0001 8.95    107.87 0.0012 -3.48  65
       90      200    0.0001 8.62    116.58 0.0012 -3.24  70
      90      200    0.0001 8.62    116.58 0.0012 -3.24  70
       25      110    0.0161 5.88    307.45 0.095  -1.98  100
      25      110    0.0161 5.88    307.45 0.095  -1.98  100
       25      110    0.0161 5.88    307.45 0.095  -1.98  100
       242     550    0.00313       7.97     647.83      0.0151 -2.68  220];
data = [data1;data2];
%==================Generator Limits ===============================
Pmin=data(:,1);
Pmax=data(:,2);
Wmax = 2000;
Wmin = 1;
% ===========Fuel cost coefficients================================
a=data(:,3);
b=data(:,4);
c=data(:,5);
% ======================== Emission Coefficients====================
d=data(:,6);
e=data(:,7);
f=data(:,8);
% ==============Transmission loss coefficients======================
B00=0.56;
B01=10^-4.*[-3.908  -1.297 7.047  0.591  2.161  -6.635 -3.908 -1.297 7.047...

            0.591   2.161  -6.635 -3.908 -1.297 7.047  0.591  2.161  -6.635...

            -3.908  -1.297 7.047  0.591  2.161  -6.635 -3.908 -1.297 7.047...

            0.591   2.161  -6.635 -3.908 -1.297 7.047  0.591  2.161  -6.635...
            -3.908  -1.297 7.047  0.591]';



B1_half=[17  12     7      -1     -5     -2     17     12     7      -1     -5
      -2     17     12     7      -1     -5     -2     17     12
       12    14     9      1      -6     -1     12     14     9      1      -6
      -1     12     14     9      1      -6     -1     12     14
       7     9      31     0      -10    -6     7      9      31     0      -10
      -6     7      9      31     0      -10    -6     7      9
       -1    1      0      24     -6     -8     -1     1      0      24     -6
      -8     -1     1      0      24     -6     -8     -1     1
       -5 -6  -10   -6     129    -2     -5     -6     -10    -6     129    -2
      -5     -6     -10    -6     129    -2     -5     -6
       -2    -1     -6     -8     -2     150    -2     -1     -6     -8     -2
      150    -2     -1     -6     -8     -2     150    -2     -1
       17    12     7      -1     -5     -2     17     12     7      -1     -5
      -2     17     12     7      -1     -5     -2     17     12
```

```
 12   14   9    1    -6   -1   12   14   9    1    -6
-1    12   14   9    1    -6   -1   12   14
 7    9    31   0    -10  -6   7    9    31   0    -10
-6    7    9    31   0    -10  -6   7    9
 -1   1    0    24   -6   -8   -1   1    0    24   -6
-8    -1   1    0    24   -6   -8   -1   1
 -5  -6-10 -6   129  -2   -5   -6   -10  -6   129  -2
-5    -6   -10  -6   129  -2   -5   -6
 -2   -1   -6   -8   -2   150  -2   -1   -6   -8   -2
150   -2   -1   -6   -8   -2   150  -2   -1
 17   12   7    -1   -5   -2   17   12   7    -1   -5
-2    17   12   7    -1   -5   -2   17   12
 12   14   9     1   -6   -1   12   14   9    1    -6
-1    12   14   9    1    -6   -1   12   14
 7    9    31   0    -10 -6 7    9    31   0    -10  -6
7     9    31   0    -10  -6   7    9
  -1  1    0    24   -6   -8   -1   1    0    24   -6
-8    -1   1    0    24   -6   -8   -1   1
  -5  -6   -10 -6 129    -2   -5   -6   -10  -6   129  -2
-5    -6   -10  -6   129  -2   -5   -6
  -2  -1   -6   -8   -2   150  -2   -1   -6   -8   -2
150   -2   -1   -6   -8   -2   150  -2   -1
 17   12   7    -1   -5   -2   17   12   7    -1   -5
-2    17   12   7    -1   -5   -2   17   12
 12   14   9    1    -6   -1   12   14   9    1    -6
-1    12   14   9    1    -6   -1   12   14];

B2_half=[7    -1   -5   -2   17   12   7    -1   -5   -2   17
12    7    -1   -5   -2   17   12   7    -1
  9   1    -6   -1   12   14   9    1    -6   -1   12
14    9    1    -6   -1   12   14   9    1
  31  0    -10  -6   7    9    31   0    -10  -6   7
9     31   0    -10  -6   7    9    31   0
  0   24   -6   -8   -1   1    0    24   -6   -8   1
1     0    24   -6   -8   -1   1    0    24
  -10 -6   129  -2   -5   -6   -10  -6   129  -2   -5
-6    -10  -6   129  -2   -5   -6   -10  -6
  -6  -8   -2   150  -2   -1   -6   -8   -2   150  -2
-1    -6   -8   -2   150  -2   -1   -6   -8
  7   -1   -5   -2   17   12   7    -1   -5   -2   17
12    7    -1   -5   -2   17   12   7    -1
  9   1    -6   -1   12   14   9    1    -6   -1   12
14    9    1    -6   -1   12   14   9    1
  31  0    -10  -6   7    9    31   0    -10  -6   7
9     31   0    -10  -6   7    9    31   0
  0   24   -6   -8   -1   1    0    24   -6   -8   -1
1     0    24   -6   -8   -1   1    0    24
  -10 -6   129  -2   -5   -6   -10  -6   129  -2   -5
-6    -10  -6   129  -2   -5   -6   -10  -6
  -6  -8   -2   150  -2   -1   -6   -8   -2   150  -2
-1    -6   -8   -2   150  -2   -1   -6   -8
  7   -1   -5   -2   17   10   -6   7    9    31   0
-10   -6   7    9    31   0    -10  -6   7
 0    24   -6   -8   -1   1    0    24   -6   -8   -1
1     0    24   -6   -8   -1   1     0   24
  -10 -6   129  -2   -5   -6   -10  -6   129  -2   -5
-6    -10  -6   129  -2   -5   -6   -10  -6
  -6  -8   12   7    -1   -5   -2   17   12   7    -1
-5    -2   17   12   7    -1  9   31  0
  9   1    -6   -1   12   14   9    1    -6   -1   12
14    9    1    -6   -1   12   14    9    1
```

```
          31    0       -2      150     -2      -1      -6      -8      -2      150     -2
         -1     -6      -8      -2      150     -2      -1      -6      -8
          7     -1      -5      -2      17      12      7       -1      -5      -2      17
         12     7       -1      -5      -2      17      12      7       -1
          9     1       -6      -1      12      14      9       1       -6      -1      12
         14     9       1       -6      -1      12      14      9       1];

B3_half = [ 7       9       31      0       -10     -6      7       9       31      0
         -10    -6      7       9       31      0       -10     -6      7       9
                -1      1       0       24      -6      -8      -1      1       0       24
         -6     -8      -1      1       0       24      -6      -8      -1      1
                -5      -6      -10     -6      129     -2      -5      -6      -10     -6
         129    -2      -5      -6      -10     -6      129     -2      -5      -6
                -2      -1      -6      -8      -2      150     -2      -1      -6      -8
         -2     150     -2      -1      -6      -8      -2      150     -2      -1
                17      12      7       -1      -5      -2      17      12      7       -1
         -5     -2      17      12      7       -1      -5      -2      17      12
                12      14      9       1       -6      -1      12      14      9       1
         -6     -1      12      14      9       1       -6      -1      12      14
                79      31      0       -10     -6      7       9       31      0       -10
         -6     7       9       31      0       -10     -6      7       9
                -1      1       0       24      -6      -8      -1      1       0       24
         -6     -8      -1      1       0       24      -6      -8      -1      1
                -5      -6      -10     -6      129     -2      -5      -6      -10     -6
         129    -2      -5      -6      -10     -6      129     -2      -5      -6
                -2      -1      -6      -8      -2      150     -2      -1      -6      -8
         -2     150     -2      -1      -6      -8      -2      150     -2      -1
                17      12      7       -1      -5      -2      17      12      7       -1
         -5     -2      17      12      7       -1      -5      -2      17      12
                12      14      9       1       -6      -1      12      14      9       1
         -6     -1      12      14      9       1       -6      -1      12      14
                79      31      0       -10     -6      7       9       31      0       -10
         -6     7       9       31      0       -10     -6      7       9
                -1      1       0       24      -6      -8      -1      1       0       24
         -6     -8      -1      1       0       24      -6      -8      -1      1
                -5      -6      -10     -6      129     -2      -5      -6      -10     -6
         129    -2      -5      -6      -10     -6      129     -2      -5      -6
                -2      -1      -6      -8      -2      150     -2      -1      -6      -8
         -2     150     -2      -1      -6      -8      -2      150     -2      -1
                17      12      7       -1      -5      -2      17      12      7       -1
         -5     -2      17      12      7       -1      -5      -2      17      12
                12      14      9       1       -6      -1      12      14      9       1
         -6     -1      12      14      9       1       -6      -1      12      14
                79      31      0       -10     -6      7       9       31      0       -10
         -6     7       9       31      0       -10     -6      7       9
                -1      1       0       24      -6      -8      -1      1       0       24
         -6     -8      -1      1       0       24      -6      -8      -1      1];

B4_half =[  31      0       -10     -6      7       9       31      0       -10     -6
         7      9       31      0       -10     -6      7       9       31      0
                024     -6      -8      -1      1       0       24      -6      -8      -1
         1      0       24      -6      -8      -1      1       0       24
               -10      -6      129     -2      -5      -6      -10     -6      129     -2
         -5     -6      -10     -6      129     -2      -5      -6      -10     -6
                -6      -8      -2      150     -2      -1      -6      -8      -2      150
         -2     -1      -6      -8      -2      150     -2      -1      -6      -8
                7-1     -5      -2      17      12      7       -1      -5      -2      17
         12     7       -1      -5      -2      17      12      7       -1
                91      -6      -1      12      14      9       1       -6      -1      12
         14     9       1       -6      -1      12      14      9       1
                31      0       -10     -6      7       9       31      0       -10     -6
         7      9       31      0       -10     -6      7       9       31      0
```

```
        024     -6     -8     -1     1      0      24     -6     -8     -1
1       0       24     -6     -8     -1     1      0      24
        -10     -6     129    -2     -5     -6     -10    -6     129    -2
-5      -6      -10    -6     129    -2     -5     -6     -10    -6
        -6      -8     -2     150    -2     -1     -6     -8     -2     150
-2      -1      -6     -8     -2     150    -2     -1     -6     -8
        7-1     -5     -2     17     12     7      -1     -5     -2     17
12      7       -1     -5     -2     17     12     7      -1
        91      -6     -1     12     14     9      1      -6     -1     12
14      9       1      -6     -1     12     14     9      1
        31      0      -10    -6     7      9      31     0      -10    -6
7       9       31     0      -10    -6     7      9      31     0
        024     -6     -8     -1     1      0      24     -6     -8     -1
1       0       24     -6     -8     -1     1      0      24
        -10     -6     129    -2     -5     -6   -10      -6     129    -2
-5      -6      -10    -6     129    -2     -5     -6     -10    -6
        -6      -8     -2     150    -2     -1     -6     -8     -2     150
-2      -1      -6     -8     -2     150    -2     -1     -6     -8
        7-1     -5     -2     17     12     7      -1     -5     -2     17
12      7       -1     -5     -2     17     12     7      -1
        91      -6     -1     12     14     9      1      -6     -1     12
14      9       1      -6     -1     12     14     9      1
        31      0      -10    -6     7      9      31     0      -10    -6
7       9       31     0      -10    -6     7      9      31     0
        024     -6     -8     -1     1      0      24     -6     -8     -1
1       0       24     -6     -8     -1     1      0      24];
B= 10^-6.*[B1_half B2_half;B3_half B4_half ];
%================Parameters for wind Power calculations=============
data =[9.891 10.785 6.861 14.482 12.184 4.602 7.232 8.036 12.333 11.074
10.040];
v0= data';
vbins=0:0.5:max(data)+ 1;
%=================weibull data
distribution=================================.
pdwind = fitdist(v0, 'weibull');
prated=11.429*10^6;
vin=5;
vr=15;
vout=45;
%===========================wind velocity constraint=====================
pwvbins = zeros(size(vbins));
powercurve = (vbins>vin)&(vbins<vr);
pwvbins(powercurve) = prated*(vbins(powercurve).^2 - vin^2)./(vr^2-vin^2);
pwvbins(vbins>=vr&vbins<=vout)=prated;
%============================wind power calculation======================
pdfv=pdf(pdwind,vbins);
avgpower=trapz(vbins,pdfv.*pwvbins);
pwd=(avgpower/(10^6));
nWindunits =input('Enter the number of available wind generators = ');
if isempty(nWindunits)|| nWindunits < 0
nWindunits = 0;
end
TotalwindPower  =nWindunits*pwd;% Total wind Power from available wind
generators
fprintf('\n\n=============================================================='
);
fprintf('\n The average wind Power available is = %8.3f
Megawatts\n',TotalwindPower);
fprintf('=============================================================');
TotalwindPower = max(Wmin,min(TotalwindPower,Wmax));
%% Initial variables of the PSO algorithm
dim = n; lb = Pmin; ub = Pmax;
```

```
vmax = 0.5*(ub-lb); vmin = -vmax;
X =zeros(num_particles,dim); V = zeros(num_particles,dim);
pbest = X; pbestVal =inf(num_particles,1); gbest = zeros(1,dim);gbestVal
=inf;
for i=1:num_particles
X(i,:)=(lb+(ub-lb).*rand(dim,1))';
V(i,:)= (vmin+(vmax-vmin).*rand(dim,1))';

[val,~]=objective(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyEm
issionFactor,powerBalancePenalty);
pbest(i,:)=X(i,:);
pbestVal(i)=val;
if val<gbestVal
  gbest=X(i,:);
  gbestVal = val;
end
end
% ==================Main loop of the PSO algorithm======================
for iter =1:maxIter
for i=1:num_particles
  r1 =rand(1,dim);r2 =rand(1,dim);
  V(i,:)=w*V(i,:)+c1*r1.*(pbest(i,:)-X(i,:))+c2*r2.*(gbest-X(i,:));
  V(i,:)=max(min(V(i,:),vmax'),vmin');
  X(i,:)=X(i,:)+V(i,:);
  X(i,:)=max(min(X(i,:),ub'),lb');

[val,~]=objective(X(i,:)',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,penaltyEm
issionFactor,powerBalancePenalty);
  if val<pbestVal(i), pbestVal(i)= val; pbest(i,:)= X(i,:);end
  if val<gbestVal,gbestVal = val; gbest = X(i,:);
  end
if mod(iter,printInterval)==0
  fprintf('Iter %d CEED=%.3f\n',iter,gbestVal);
end
end
end
%==============================Final evaluation of
variables==============================
[Pcost,details]=objective(gbest',PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,pe
naltyEmissionFactor,powerBalancePenalty);

fprintf('\n==========Final PSO Dispatch (6units)===========\n');
for i=1:n
    fprintf(' P(%d)=%.3f MW (min %.1f max
%.1f)\n',i,gbest(i),Pmin(i),Pmax(i))
end
fprintf('Sum Thermal=%.3f MW, Wind=%.3f MW, Loss=%.3f
MW\n',sum(gbest),TotalwindPower,details.PL)
fprintf('Fuel Cost=%.3f , Emission=%.3f ,
CEED=%.3f\n',details.fuelcost,details.emission,Pcost);
toc
% This function is called on the main program: WTEED_casePSO40units for
calculation
% of PL,fuelcost,emission.Named: objective.m
function[cost,details]=objective(P,PD,TotalwindPower,a,b,c,d,e,f,B00,B01,B,
k1,kpb)
P =P(:); WG = P+(TotalwindPower/length(P));
PL = B00+B01'*WG+WG'*B*WG;
fuel=sum(a.*(P.^2)+b.*P+c);
em =sum(d.*(P.^2)+e.*P+f);
CEED = fuel+k1*em;
mismatch=PD+PL-(sum(P)+TotalwindPower);
```

```
penalty=kpb*mismatch^2;
cost =CEED+penalty;
details.PL=PL;details.fuelcost=fuel;details.emission=em;
end
%===========================================================================
==
```

## APPENDIX C: DEVELOPED MATLAB PROGRAM FOR MULTI-AREA WTEED PROBLEM USING LAGRANGE'S ALGORITHM

### APPENDIX C1: MATLAB script file – MAWTEED_case6units.m

```
% M-File : MAWTEED_case6units.m
% two-area and 3 generator in each area
% Transmission loss considerd in this case
% 1 tie-lines are used to interconnect the two-area power system
% ==================M-File Description==============================
% The M-File is used to solve Multiarea economic emission dispatch problem
by Lagrange's Algorithm
% The software is developed for MAWTEED problem. This program ised to
% create table 5.1, and also used in table 5.2.
clearvars
clc
%% Multi area economic dispatch data taken from the reference paper
(Basu.,2013 and Alli.,2024)
tic
%% Multi area Wind thermal economic emission dispatch data
m=2; % No of area
n=6; % Total No of Generators in all the area
NG=3; % No of Generators in each area
TL=1; % NO of Tie lines
N_iter=1000; % No of iteration
lambda=[50 50];
lambda1=lambda(1);lambda2=lambda(2);
lambda={lambda(1),lambda(2)};
PD=[757.8 505.2]; % Area power demand
%PD=[525 435 605 625]; % Area power demand
alfa=[0.01 0.01]; %Incremental deltalambda Tolerance value of each area
epsilon=[0.01 0.01]; % Tolerance value of each area
%%============Multi area Generator fuel Cost coefficients==============
c=[550 350 310 240 200 126];
c={c(1:3),c(4:6)};
b=[8.10 7.50 8.10 7.74 8.00 8.60];
b={b(1:3),b(4:6)};
a = [0.00028 0.00056 0.00056 0.00324 0.00254 0.00284];
a={a(1:3),a(4:6)};
%%============Multi area Emission coefficients=========================
d = [0.00683 0.00461 0.00461 0.00484 0.00754 0.00661];
d={d(1:3),d(4:6)};
e=[-0.54551 -0.51160 -0.51160 -0.32767 -0.54551 -0.63262];
e={e(1:3),e(4:6)};
f = [40.26690 42.89553 42.89553 33.85932 50.639310 45.83267];
f={f(1:3),f(4:6)};
%%===========Multi area Generator real power limits====================
Pmin = [100 50 50 80 50 50];
Pmin1={Pmin(1:3),Pmin(4:6)};
Pmax =[500 200 150 300 200 120];
Pmax1={Pmax(1:3),Pmax(4:6)};
%%================Intial Tie line values (Assumed)=====================
PT_mj=[10];
PT_jm=[15];
%%=====================Tie line limits================================
PTmin_mj=[5];
```

```
PTmin_jm=[5];
PTmax_mj=[100];
PTmax_jm=[100];
%%=============Tie line coefficients====================================
q_mj=rand(1,1);q_jm=rand(1,1);
%%=============Tie line Fractional loss rate values=====================
flr_jm=[0.11];
%%====================Tie line incremental value========================
alpha_mj=[0.000001];
alpha_jm=[0.00001];
%%====================Transmission loss coefficients====================
B_area=[ 0.000071 0.00003 0.000025
         0.00003 0.000069 0.000032
         0.000025 0.000032 0.00008
   0.000056 0.000045 0.000015
   0.000023 0.000042 0.000047
   0.000032 0.000023 0.000027];
B_area1={B_area((1:3),:),B_area((4:6),:)};
% ===================Wind power data for each area=================
windData = {[9.891 10.785 6.861 14.482 12.184 4.602]...
            [8.5 7.2 6.9 10.4 12.1 5.8]};

Prated = [ 555 0]; %Rated wind power per area in (MW)
Vin = 3; Vr = 12; Vout = 25;
vbins = 0:0.5:30;
Ns = 50; %Number of scenarios
TotalwindPower = zeros(m,Ns);
for area = 1:m
v0 = windData{area}';
    pwind = fitdist(v0,'Weibull');
    %=============== Power for each area================================
     pwvbins = zeros(size(vbins));
     pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
     ((vbins(vbins>=Vin &vbins<Vr).^2 - Vin^2) /(Vr^2 - Vin^2));
     pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
  % ================Expected wind power=================================
     pdfv = pdf(pwind,vbins);
     avgpower = trapz(vbins,pdfv .*pwvbins);
    fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
    for s =1:Ns
        v = random(pwind);
        if v<Vin
        Pw = 0;
        elseif v < Vr
        Pw = Prated(area) * ((v^2 - Vin^2)/(Vr^2 - Vin^2));
        elseif v < Vout
        Pw = Prated(area);
        else
        Pw = 0;
        end
      TotalwindPower(area,s) = Pw;
      end
      end
      %====================Expected wind power
      ==============================
      Pwind_exp = mean(TotalwindPower,2);
      % ==================CEED with wind
      power====================================
      Pwind = Pwind_exp; % Expected wind power
      % Generator real power calculation
      %% ================Lambda values for 2
      areas============================
```

```
        lambda = [lambda1, lambda2];
        %%==================WTEED=========================================
        for iter = 1:N_iter
       fprintf('Iter %d\n',iter);
        %=========Caculation of generator power per
        area=======================
     Power1 =zeros(m,3); %Pre allocation of variables
     h1 =zeros(m,3);
     for i =1:m
        %============Calculation of the penalty factor h using max/max
=======

h=(a{i}.*Pmax1{i}.^2+b{i}.*Pmax1{i}+c{i})./(d{i}.*Pmax1{i}.^2+e{i}.*Pmax1{i
}+f{i});
  E = diag((a{i}+h.*d{i}) ./lambda(i));
  D = 0.5*(1-(b{i}+h.*e{i})./lambda(i))';
  Power=E\D;
  Power1(i,:)=Power';
  h1(i,:)=h;
end
P=reshape(Power1',1,6);
h=reshape(h1',1,6);
h_cell ={h(1:3),h(4:6)};

%%% =================Check generator limits=======================
for i = 1:n

  P(i) = min(max(P(i), Pmin(i)),Pmax(i));
end
%%======================Tie line power flow============================
ePT_mj=q_mj+lambda1;
PT_mj=(PT_mj-alpha_mj.*ePT_mj);
ePT_jm=q_jm-(1-flr_jm).*lambda1 ;
PT_jm=PT_jm-alpha_jm.*ePT_jm;
%%======================Tie Line limits================================
TL=1;
for i=1:TL
PT_mj(i)=min(max(PT_mj(i),PTmin_mj(i)),PTmax_mj(i));
PT_jm(i)=min(max(PT_jm(i),PTmin_jm(i)),PTmax_jm(i));
end
Tielinepower=PT_mj-(1-flr_jm).*PT_jm;
%%=====================Tie line flows per area net
injection===============
% Tieline_area1 = Tielinepower(1) + Tielinepower(2) + Tielinepower(3);
Tieline_area1 = Tielinepower;
Tieline_area2 = -Tielinepower;
Tieline_area = [Tieline_area1, Tieline_area2];
%%=======================Transmission line loss=======================
B_area11=permute(reshape(B_area,3,2,[]),[1 3 2]);
P=reshape(P,3,1,[]);
PL_area=bsxfun(@times,P,reshape(P,1,3,[])).*B_area11;
PL=sum(reshape(PL_area,[],2));
%%==================Calculation of delta using P sums per area======
P1 = P(1:3);P2 = P(4:6);
sumP = [sum(P1),sum(P2)];
deltalambda = ( PD + PL + Tieline_area)-(sumP+Pwind');
%%=========Updating lambda using gradient procedure===================
    lambda = lambda + alfa.*deltalambda;
  %%================Breaking upon convergence and reaching max
iterations===
     if all(abs(deltalambda)<=epsilon)
```

```
        fprintf('Converged in %d iterations (all |deltalambda| <=
epsilon).\n',iter);
        break;
     elseif iter>= N_iter
        fprintf('Reached maximum iterations %d (not all area
converged).\n',N_iter)
        break;
     end
 end
P1 = P(1:3);P2 = P(4:6);
h = h_cell;
% lambda = {lambda1, lambda2};
lambda
P
Tielinepower;
PT_mj;
PT_jm;
P={P(1:3),P(4:6)};
%=== Calculation of fuel cost, emission and CEED values of MAEED problem
solution===
for i=1:m
fuelcost_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i});
fuelcost(i,:)=fuelcost_area;
emissioncost_area=sum(d{i}.*P{i}.^2+e{i}.*P{i}+f{i});
emissioncost(i,:)=emissioncost_area;
ceed_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i}+h{i}.*(d{i}.*P{i}.^2+e{i}.*P{i}
+f{i}));
ceed(i,:)=ceed_area;
end
fuelcost=sum(fuelcost)
emissioncost=sum(emissioncost)
ceed=sum(ceed)
ans=[fuelcost emissioncost ceed iter toc]
fprintf('Final Wind Injection per Area (MW):\n')
disp(Pwind');
fprintf('Final results after %d iterations\n',iter);
fprintf('Total Fuel cost = %.4f\n',fuelcost);
fprintf('Total Emission cost = %.4f\n',emissioncost)
fprintf('Total CEED = %.4f\n',ceed);
fprintf('Final Wind Injection per Area (MW):\n');disp(Pwind' );
toc
```

**APPENDIX C2:** **MATLAB script file – MAWTEED_case12units.m**

```
% M-File : MAWTEEDP_case12units.m
% =================M-File Description==================================
% This test system consists of a standard IEEE30 bus system with four-area
% ,12 thermal generators, comprising 3 generators in each area,
% and a three-wind park network that supplies
% wind power of 110MW each to the entire dispatch problem.
% The total power demand is considered as 2090[MW].
% The software is developed for MAWTEED problem. The program is used to
% develop table 5.3 and used in table 5.4
clearvars;
clc
%% Multi area Wind thermal economic emission dispatch data
m=4; % No of area
n=12; % Total No of Generators in all the area
NG=3; % No of Generators in each area
TL=6; % NO of Tie lines
N_iter=20000; % No of iteration
lambda=[100 100 100 100];
```

```
lambda1=lambda(1);lambda2=lambda(2);lambda3=lambda(3);lambda4=lambda(4);
lambda={lambda(1),lambda(2),lambda(3),lambda(4)};
PD=[500 410 580 600]; % Area power demand
%PD=[525 435 605 625]; % Area power demand
alfa=[0.01 0.01 0.01 0.01]; %Incremental deltalambda Tolerance value of
each area
epsilon=[0.01 0.01 0.01 0.01]; % Tolerance value of each area
%%============Multi area Generator fuel Cost coefficients=============
a=[0.03546 0.02111 0.01799...
   0.15247 0.02803 0.14834...
   0.10587 0.07505 0.11934...
   0.10587 0.13552 0.08963];
a={a(1:3),a(4:6),a(7:9),a(10:12)};

b=[38.30553 36.32782 38.27041...
   38.53973 40.39655 38.34001...
   46.15916 43.83562 50.63211...
   46.15916 41.03782 33.56211];
b={b(1:3),b(4:6),b(7:9),b(10:12)};

c=[1243.5311 1658.5696 1356.6592...
   0756.7989 0449.9977 0558.5696...
   0451.3251 0673.0267 0530.7199...
   0851.3251 1038.533 1285.907];
c={c(1:3),c(4:6),c(7:9),c(10:12)};
%%============Multi area Emission coefficients=======================
d=[0.00683 0.00461 0.00461...
   0.00484 0.00754 0.00661...
   0.00914 0.00533 0.00674...
   0.00728 0.00479 0.00387];
d={d(1:3),d(4:6),d(7:9),d(10:12)};

e=[-0.54551 -0.51160 -0.51160...
   -0.32767 -0.54551 -0.63262...
   -0.43211 -0.61173 -0.49731...
   -0.6821 -0.50660 -0.49340];
e={e(1:3),e(4:6),e(7:9),e(10:12)};

f=[40.26690 42.89553 42.89553...
   33.85932 50.639310 45.83267...
   48.21560 52.45210 41.10420...
   30.36320 25.17650 27.75490];
f={f(1:3),f(4:6),f(7:9),f(10:12)};
Wmin = 1;
Wmax = 2000;
nW = 1;
%%===========Multi area Generator real power limits===================
Pmin=[35 130 125 10 35 125 15 30 50 15 30 50];
Pmin1={Pmin(1:3),Pmin(4:6),Pmin(7:9),Pmin(10:12)};
Pmax=[210 325 315 150 110 215 175 215 335 175 215 335];
Pmax1={Pmax(1:3),Pmax(4:6),Pmax(7:9),Pmax(10:12)};
%%===============Intial Tie line values (Assumed)=====================
PT_mj=[10 15 12 20 18 29];
PT_jm=[15 18 20 14 22 19];
%%====================Tie line limits================================
PTmin_mj=[5 5 5 5 5 5];
PTmin_jm=[5 5 5 5 5 5];
PTmax_mj=[60 50 60 60 60 50];
PTmax_jm=[50 60 60 60 50 60];
%%============Tie line coefficients===================================
q_mj=rand(1,6);q_jm=rand(1,6);
%%============Tie line Fractional loss rate values====================
```

```
% flr_jm=[0.11 0.21 0.14 0.16];
flr_jm=[0.11 0.21 0.14 0.16 0.22 0.11];
%%====================Tie line incremental value========================
alpha_mj=[0.000001 0.000001 0.000001 0.000001 0.000001 0.000001];
% alpha_mj=[0.000001 0.000001 0.000001 0.000001];
alpha_jm=[0.00001 0.00001 0.00001 0.00001 0.00001 0.00001];
% alpha_jm=[0.00001 0.00001 0.00001 0.00001];
%%====================Transmission loss coefficients===================
B_area=[ 0.000071 0.00003 0.000025
         0.00003 0.000069 0.000032
         0.000025 0.000032 0.00008
         0.000056 0.000045 0.000015
         0.000023 0.000042 0.000047
         0.000032 0.000023 0.000027
  0.00002 0.000028 0.000053
  0.000086 0.000034 0.000016
  0.000053 0.000016 0.000028
  0.000074 0.00003 0.000025
  0.000049 0.000069 0.000037
  0.000022 0.000032 0.000083];
B_area1={B_area((1:3),:),B_area((4:6),:),B_area((7:9),:),B_area((10:12),:)}
;
% ====================Wind power data for each area=================
windData = {[9.891 10.785 6.861 14.482 12.184 4.602];...
            [8.5 7.2 6.9 10.4 12.1 5.8];...
            [7.3 8.6 6.2 11.0 9.8 4.5];...
            [6.5 7.9 5.1 9.2 8.7 4.0];};
Prated = [ 215 210 210 212]; %Rated wind power per area in (MW)
Vin = 3; Vr = 12; Vout = 25;
vbins = 0:0.5:30;
Ns = 50; %Number of scenarios
TotalwindPower = zeros(m,Ns);
for area = 1:m
    v0 = windData{area}';
    pwind = fitdist(v0,'Weibull');
    %=============== Power for each area================================
     pwvbins = zeros(size(vbins));
     pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
     ((vbins(vbins>=Vin &vbins<Vr).^2 - Vin^2) /(Vr^2 - Vin^2));
     pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
  % ================Expected wind power================================
     pdfv = pdf(pwind,vbins);
     avgpower = trapz(vbins,pdfv .*pwvbins);
    fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
    for s =1:Ns
        v = random(pwind);
        if v<Vin
        Pw = 0;
        elseif v < Vr
        Pw = Prated(area) * ((v^3 - Vin^3)/(Vr^3 - Vin^3));
        elseif v < Vout
        Pw = Prated(area);
        else
        Pw = 0;
        end
  TotalwindPower(area,s) = Pw;
    end
end
%=====================Expected wind power =============================
Pwind_exp = mean(TotalwindPower,2);
% ==================CEED with wind power===============================
Pwind = Pwind_exp; % Expected wind power
```

```
%================Generator real power calculation======================
for iter=1:N_iter
iter;
for i=1:m
%h=(a{i}.*Pmin1{i}.^2+b{i}.*Pmin1{i}+c{i})./(d{i}.*Pmax1{i}.^2+e{i}.*Pmax1{
i}+f{i});
h=(a{i}.*Pmax1{i}.^2+b{i}.*Pmax1{i}+c{i})./(d{i}.*Pmax1{i}.^2+e{i}.*Pmax1{i
}+f{i});
%h=(a{i}.*Pmin1{i}.^2+b{i}.*Pmin1{i}+c{i})./(d{i}.*Pmin1{i}.^2+e{i}.*Pmin1{
i}+f{i});
%h=(a{i}.*Pmax1{i}.^2+b{i}.*Pmax1{i}+c{i})./(d{i}.*Pmin1{i}.^2+e{i}.*Pmin1{
i}+f{i});
E=diag((a{i}+h.*d{i})./lambda{i})+B_area1{i};
D=0.5*(1-(b{i}+h.*e{i})./lambda{i})';
Power=E\D;
Power1(i,:)=Power;
h1(i,:)=h;
end
P=reshape(Power1',1,12);
h=reshape(h1',1,12);
h={h(1:3),h(4:6),h(7:9),h(10:12)};
%%==================Generator Real powers==============================
P=[P(1:3),P(4:6),P(7:9),P(10:12)];
%%==================Generator real power limits=========================
for i=1:n % Total Number of generators in all the four area is n=12
 P(i)=min(max(P(i),Pmin(i)),Pmax(i));
end
%%======================Tie line power flow=============================
ePT_mj=q_mj+lambda1;PT_mj=(PT_mj-alpha_mj.*ePT_mj);
ePT_jm=q_jm-(1-flr_jm).*lambda1 ;PT_jm=PT_jm-alpha_jm.*ePT_jm;
%%======================Tie Line limits=================================
TL=6;
for i=1:TL

 PT_mj(i)=min(max(PT_mj(i),PTmin_mj(i)),PTmax_mj(i));
 PT_jm(i)=min(max(PT_jm(i),PTmin_jm(i)),PTmax_jm(i));
end
Tielinepower=PT_mj-(1-flr_jm).*PT_jm;
Tielinepower_area = (sum(Tielinepower) / m) * ones(1,m);
%%========================Transmission line loss========================
B_area11=permute(reshape(B_area,3,4,[]),[1 3 2]);
P=reshape(P,3,1,[]);
PL_area=bsxfun(@times,P,reshape(P,1,3,[])).*B_area11;
PL=sum(reshape(PL_area,[],4));
%%==================Incremental lambda==================================
P1=P(1:3);P2=P(4:6);P3=P(7:9);P4=P(10:12);
deltalambda1=(PD(1)+PL(1)+Tielinepower(1))-(sum(P1)+Pwind(1));
deltalambda2=(PD(2)+PL(2)+Tielinepower(2))-(sum(P2)+Pwind(2));
deltalambda3=(PD(3)+PL(3)+Tielinepower(3))-(sum(P3)+Pwind(3));
deltalambda4=(PD(4)+PL(4)+Tielinepower(4))-(sum(P4)+Pwind(4));
deltalambda=[deltalambda1 deltalambda2 deltalambda3 deltalambda4];
if abs(deltalambda1)<=epsilon(1) && iter>=N_iter
     fprintf('Converged in %d iterations (all |deltalambda| <=
epsilon).\n',iter);
break
else
    lambda1=lambda1+(deltalambda1*alfa(1));
end
if abs(deltalambda2)<=epsilon(2) && iter>=N_iter
break
else lambda2=lambda2+(deltalambda2*alfa(2));
end
```

```
if abs(deltalambda3)<=epsilon(3) && iter>=N_iter
break
else lambda3=lambda3+(deltalambda3*alfa(3));
end
if abs(deltalambda4)<=epsilon(4) && iter>=N_iter
break
else lambda4=lambda4+(deltalambda4*alfa(4));
end
lambda={lambda1,lambda2,lambda3,lambda4};
end
deltalambda=[deltalambda1 deltalambda2 deltalambda3 deltalambda4];
P
PL
Tielinepower
PT_mj
PT_jm
P={P(1:3),P(4:6),P(7:9),P(10:12)}
for i=1:m
fuelcost_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i});
fuelcost(i,:)=fuelcost_area;
emissioncost_area=sum(d{i}.*P{i}.^2+e{i}.*P{i}+f{i});
emissioncost(i,:)=emissioncost_area;
ceed_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i}+h{i}.*(d{i}.*P{i}.^2+e{i}.*P{i}
+f{i}));
ceed(i,:)=ceed_area;
end
fuelcost=sum(fuelcost)
emissioncost=sum(emissioncost)
ceed=sum(ceed)
% fprintf('Final Wind Injection per Area (MW):\n');
% disp(Pwind');

fprintf('Final results after %d iterations\n', iter);
fprintf('Total Fuel cost = %.4f\n', fuelcost);
fprintf('Total Emission cost = %.4f\n', emissioncost);
fprintf('Total CEED = %.4f\n', ceed);
fprintf('Final Wind Injection per Area (MW):\n'); disp(Pwind');
Pwind = sum(Pwind(:,:))
PL = sum(PL(:,:))
```

### **APPENDIX C3:** MATLAB script file – MAWTEED_case40units.m

```
% M-File : MAWTEED_case40units.m
% ==================M-File Description===================================
% two-area and 20 generator in each area. Transmission loss is omitted in
this case
% 1 tie-lines are used to interconnect the two-area power system
% a three-wind park network that supplies wind power of 110MW each to the
entire dispatch problem.
% The total power demand is considered as 10500[MW].
% The software is developed for MAWTEED problem. The program is used to
% develop table 5.9 and used in table 5.10
% ==================M-File Description===================================
clearvars
clc
%% Multi area economic dispatch data taken from the reference paper
(Basu,2011)
tic
m=2; % No of area
n=40; % Total No of Generators in all the area
TL=1; % NO of Tie lines
N_iter=50000; % No of iteration
```

```
lambda=[100 100]; % Initial Lagrangian variable
lambda1=lambda(1);lambda2=lambda(2);
lambda={lambda(1),lambda(2)};
% Total power demand of the system, PD=10500[MW]
PD=[7500 3000]; % Area power demand
alfa=[0.01 0.01]; %Incremental deltalambda Tolerance value of each area
epsilon=[0.01 0.01]; % Tolerance value of each area
%% Multi area Generator fuel Cost coefficients
%% Single area Generator fuel Cost coefficients
%% ===========Data are given in the reference paper (Basu, 2013)==========
%The data are given in the format[GenNum Pmin Pmax a b c d e f]============
Data=[.....
1 36 114 0.0069  6.73 94.705 0.048  -2.22  60
2 36 114 0.0069  6.73 94.705 0.048  -2.22  60
3 60 120 0.02028 7.07 309.54 0.0762 -2.36  100
4 80 190 0.00942 8.18 369.03 0.054  -3.14  120
5 47 97  0.0114  5.35 148.89 0.085  -1.89  50
6 68 140 0.01142 8.05 222.33 0.0854 -3.08  80
7 110 300 0.00357 8.03 287.71 0.0242 -3.06 100
8 135 300 0.00492 6.99 391.98 0.031  -2.32 130
9 135 300 0.00573 6.6 455.76 0.0335 -2.11 150
10 130 300 0.00605 12.9 722.82 0.425 -4.34 280
11 94 375 0.00515 12.9 635.2 0.0322 -4.34 220
12 94 375 0.00569 12.8 654.69 0.0338 -4.28 225
13 125 500 0.00421 12.5 913.4 0.0296 -4.18 300
14 125 500 0.00752 8.84 1760.4 0.0512 -3.34 520
15 125 500 0.00752 8.84 1760.4 0.0496 -3.55 510
16 125 500 0.00752 8.84 1760.4 0.0496 -3.55 510
17 220 500 0.00313 7.97 647.85 0.0151 -2.68 220
18 220 500 0.00313 7.95 649.69 0.0151 -2.66 222
19 242 550 0.00313 7.97 647.83 0.0151 -2.68 220
20 242 550 0.00313 7.97 647.81 0.0151 -2.68 220
21 254 550 0.00298 6.63 785.96 0.0145 -2.22 290
22 254 550 0.00298 6.63 785.96 0.0145 -2.22 285
23 254 550 0.00284 6.66 794.53 0.0138 -2.26 295
24 254 550 0.00284 6.66 794.53 0.0138 -2.26 295
25 254 550 0.00277 7.1 801.32 0.0132 -2.42  310
26 254 550 0.00277 7.1 801.32 0.0132 -2.42  310
27 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
28 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
29 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
30 47 97  0.0114  5.35 148.89 0.085 -1.89   50
31 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
32 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
33 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
34 90 200 0.0001 8.95 107.87  0.0012 -3.48   65
35 90 200 0.0001 8.62 116.58  0.0012 -3.24   70
36 90 200 0.0001 8.62 116.58  0.0012 -3.24   70
37 25 110 0.0161 5.88 307.45  0.095  -1.98   100
38 25 110 0.0161 5.88 307.45  0.095  -1.98   100
39 25 110 0.0161 5.88 307.45  0.095  -1.98   100
40 242 550 0.00313 7.97 647.83 0.0151 -2.68  220];
%% =====================Multi area fuel cost coefficients================
a=Data(:,4)';
a={a(1:20),a(21:40)};
b=Data(:,5)';
b={b(1:20),b(21:40)};
c=Data(:,6)';
c={c(1:20),c(21:40)};
%%====================Multi area Emmis on coefficients===================
d=Data(:,7)';
d={d(1:20),d(21:40)};
```

```
e=Data(:,8)';
e={e(1:20),e(21:40)};
f=Data(:,9)';
f={f(1:20),f(21:40)};
%% ==============Multi area Generator real power limits================
Pmin=Data(:,2)';
Pmin1={Pmin(1:20),Pmin(21:40)};
Pmax=Data(:,3)';
Pmax1={Pmax(1:20),Pmax(21:40)};
%%============= Initial Tie line values between 100 to 200 [MW] (Assumed)
PT_mj=randi([100 200],1,1);
PT_jm=randi([100 200],1,1);
%% ======================Tie line limits (Assumed)=======================
PTmin_mj=[100];
PTmin_jm=[100];
PTmax_mj=[1500];
PTmax_jm=[1500];
%% ==================Tie line coefficients (Assumed)====================
q_mj=rand(1,1);
q_jm=rand(1,1);
%% ===========Tie line Fractional loss rate values (Assumed)=============
flr_jm=rand(1,1);
%%===================Tie line incremental value (Assumed)================
alpha_mj=[0.01];
alpha_jm=[0.01];
% ===================Wind power data for each area=================
windData = {[9.891 10.785 6.861 14.482 12.184 4.602]...
            [8.5 7.2 6.9 10.4 12.1 5.8]};
Prated = [ 555 0]; %Rated wind power per area in (MW)
Vin = 5; Vr = 15; Vout = 25;
vbins = 0:0.5:30;
Ns = 50; %Number of scenarios
TotalwindPower = zeros(m,Ns);
for area = 1:m
    v0 = windData{area}';
    pwind = fitdist(v0,'Weibull');
    %=============== Power for each area================================
     pwvbins = zeros(size(vbins));
     pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
     ((vbins(vbins>=Vin &vbins<Vr).^2 - Vin^2) /(Vr^2 - Vin^2));
     pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
  % ================Expected wind power=================================
     pdfv = pdf(pwind,vbins);
     avgpower = trapz(vbins,pdfv .*pwvbins);
    fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
    for s =1:Ns
        v = random(pwind);
        if v<Vin
        Pw = 0;
        elseif v < Vr
        Pw = Prated(area) * ((v^2 - Vin^2)/(Vr^2 - Vin^2));
        elseif v < Vout
        Pw = Prated(area);
        else
        Pw = 0;
        end
  TotalwindPower(area,s) = Pw;
    end
end
%=====================Expected wind power ==============================
Pwind_exp = mean(TotalwindPower,2);
% ===================CEED with wind power==============================
```

```
Pwind = Pwind_exp; % Expected wind power
% Generator real power calculation
%% ================Lambda values for 2 areas============================
lambda = [lambda1, lambda2];
%%===================WTEED==========================================
 for iter = 1:N_iter
     fprintf('Iter %d\n',iter);
  %=========Caculation of generator power per area=======================
    Power1 =zeros(m,20); %Pre allocation of variables
    h1 =zeros(m,20);
    for i =1:m
   %===========Calculation of the penalty factor h using max/max =======

h=(a{i}.*Pmax1{i}.^2+b{i}.*Pmax1{i}+c{i})./(d{i}.*Pmax1{i}.^2+e{i}.*Pmax1{i
}+f{i});
        E = diag((a{i}+h.*d{i}) ./lambda(i));
        D = 0.5*(1-(b{i}+h.*e{i})./lambda(i))';
        Power=E\D;
        Power1(i,:)=Power';
        h1(i,:)=h;
    end
     P=reshape(Power1',1,40);
    h=reshape(h1',1,40);
    h_cell ={h(1:20),h(21:40)};

%%% ================Check generator limits======================
    for i = 1:n
        P(i) = min(max(P(i), Pmin(i)),Pmax(i));
    end
%%=====================Tie line power flow============================
ePT_mj=q_mj+lambda1;
PT_mj=(PT_mj-alpha_mj.*ePT_mj);
ePT_jm=q_jm-(1-flr_jm).*lambda1 ;
PT_jm=PT_jm-alpha_jm.*ePT_jm;
%%=====================Tie Line limits================================
TL=1;
for i=1:TL
 PT_mj(i)=min(max(PT_mj(i),PTmin_mj(i)),PTmax_mj(i));
 PT_jm(i)=min(max(PT_jm(i),PTmin_jm(i)),PTmax_jm(i));
end
Tielinepower=PT_mj-(1-flr_jm).*PT_jm;
    %%====================Tie line flows per area net
injection================
    % Tieline_area1 = Tielinepower(1) + Tielinepower(2) + Tielinepower(3);
    Tieline_area1 = Tielinepower;
    Tieline_area2 = -Tielinepower;
    Tieline_area = [Tieline_area1, Tieline_area2];
   %%=================Calculation of delta using P sums per area======
    P1 = P(1:20);P2 = P(21:40);
    sumP = [sum(P1),sum(P2)];
    deltalambda = ( PD + Tieline_area)-(sumP+Pwind');
%%=========Updating lambda using gradient procedure====================
    lambda = lambda + alfa.*deltalambda;
  %%===============Breaking upon convergence and reaching max
iterations===
     if all(abs(deltalambda)<=epsilon)
        fprintf('Converged in %d iterations (all |deltalambda| <=
epsilon).\n',iter);
        break;
     elseif iter>= N_iter
        fprintf('Reached maximum iterations %d (not all area
converged).\n',N_iter)
```

```
        break;
      end
 end
P1 = P(1:20);P2 = P(21:40);
h = h_cell;
% lambda = {lambda1, lambda2};
lambda
P
Tielinepower;
PT_mj;
PT_jm;
P={P(1:20),P(21:40)};
%=== Calculation of fuel cost, emission and CEED values of MAEED problem
solution===
for i=1:m
fuelcost_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i});
fuelcost(i,:)=fuelcost_area;
emissioncost_area=sum(d{i}.*P{i}.^2+e{i}.*P{i}+f{i});
emissioncost(i,:)=emissioncost_area;
ceed_area=sum(a{i}.*P{i}.^2+b{i}.*P{i}+c{i}+h{i}.*(d{i}.*P{i}.^2+e{i}.*P{i}
+f{i}));
ceed(i,:)=ceed_area;
end
fuelcost=sum(fuelcost)
emissioncost=sum(emissioncost)
ceed=sum(ceed)
ans=[fuelcost emissioncost ceed iter toc]
fprintf('Final Wind Injection per Area (MW):\n')
disp(Pwind');
fprintf('Final results after %d iterations\n', iter);
fprintf('Total Fuel cost = %.4f\n', fuelcost);
fprintf('Total Emission cost = %.4f\n', emissioncost);
fprintf('Total CEED = %.4f\n', ceed);
fprintf('Final Wind Injection per Area (MW):\n'); disp(Pwind');
toc
ans1 = [P Pwind fuelcost emissioncost ceed]
lambda
```

## APPENDIX D: DEVELOPED MATLAB PROGRAM FOR MULTI-AREA WTEED PROBLEM USING PSO ALGORITHM

### APPENDIX D1: MATLAB script file – MAWTEED_casePSO6units.m

```
% M-File : MAWTEED_casePSO6units.m
% two-area and 3 generator in each area
% Transmission loss considerd in this case
% 1 tie-lines are used to interconnect the two-area power system
% ==================M-File Description===============================
% The M-File is used to solve Multiarea economic emission dispatch problem
using PSO Algorithm
% The software is developed for MAWTEED problem. This program is used to
% create table 6.1 and used in table 6.2. It is used with the function
% named: objective_funPSO.m
clearvars
clc
%% Multi area economic dispatch data taken from the reference paper
(Basu.,2013 and Alli.,2024)
tic
%% Multi area Wind thermal economic emission dispatch data
m=2; % No of area
n=6; % Total No of Generators in all the area
NG=3; % No of Generators in each area
TL=1; % NO of Tie line
```

```
PD=[757.8 505.2]; % Area power demand
%PD=[525 435 605 625]; % Area power demand
%%============Multi area Generator fuel Cost coefficients==============
c=[550 350 310 240 200 126];
% c={c(1:3),c(4:6)};
b=[8.10 7.50 8.10 7.74 8.00 8.60];
% b={b(1:3),b(4:6)};
a = [0.00028 0.00056 0.00056 0.00324 0.00254 0.00284];
% a={a(1:3),a(4:6)};
%%============Multi area Emission coefficients========================
d = [0.00683 0.00461 0.00461 0.00484 0.00754 0.00661];
% d={d(1:3),d(4:6)};
e=[-0.54551 -0.51160 -0.51160 -0.32767 -0.54551 -0.63262];
% e={e(1:3),e(4:6)};
f = [40.26690 42.89553 42.89553 33.85932 50.639310 45.83267];
% f={f(1:3),f(4:6)};
%%===========Multi area Generator real power limits====================
Pmin = [100 50 50 80 50 50];
% Pmin1={Pmin(1:3),Pmin(4:6)};
Pmax =[500 200 150 300 200 120];
% Pmax1={Pmax(1:3),Pmax(4:6)};
%%===============Intial Tie line values (Assumed)=====================
% PT_mj=10;
% PT_jm=15;
%%=====================Tie line limits=================================
% PTmin_mj=5;
% PTmin_jm=5;
% PTmax_mj=100;
% PTmax_jm=100;
%%============Tie line coefficients====================================
% q_mj=rand(1,1);q_jm=rand(1,1);
%%============Tie line Fractional loss rate values=====================
% flr_jm=0.11;
% %%===================Tie line incremental value=======================
% alpha_mj=0.000001;
% alpha_jm=0.00001;
%%===================Transmission loss coefficients===================
B_area=[ 0.000071 0.00003 0.000025
   0.00003 0.000069 0.000032
   0.000025 0.000032 0.00008
   0.000056 0.000045 0.000015
   0.000023 0.000042 0.000047
   0.000032 0.000023 0.000027];
% B_area1={B_area((1:3),:),B_area((4:6),:)};
% ===================Wind power data for each area=================
windData = {[9.891 10.785 6.861 14.482 12.184 4.602]...
            [8.5 7.2 6.9 10.4 12.1 5.8]};

Prated = [ 555 0]; %Rated wind power per area in (MW)
Vin = 3; Vr = 12; Vout = 25;
vbins = 0:0.5:30;
Ns = 50; %Number of scenarios
TotalwindPower = zeros(m,Ns);
for area = 1:m
    v0 = windData{area}';
    pwind = fitdist(v0,'Weibull');
    %=============== Power for each area===============================
     pwvbins = zeros(size(vbins));
     pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
     ((vbins(vbins>=Vin &vbins<Vr).^3 - Vin^3) /(Vr^3 - Vin^3));
     pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
  % ================Expected wind power===============================
```

```
     pdfv = pdf(pwind,vbins);
     avgpower = trapz(vbins,pdfv .*pwvbins);
    fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
for s =1:Ns
  v = random(pwind);
  if v<Vin
  Pw = 0;
  elseif v < Vr
  Pw = Prated(area) * ((v^3 - Vin^3)/(Vr^3 - Vin^3));
  elseif v < Vout
  Pw = Prated(area);
  else
  Pw = 0;
  end
TotalwindPower(area,s) = Pw;
end
end
%=====================Expected wind power ==============================
Pwind_exp = mean(TotalwindPower,2);
% ===================CEED with wind power==============================
Pwind = Pwind_exp; % Expected wind power
%%===================WTEED=========================================
%% ========================Price penalty factor h=======================
h = (a.*Pmax.^2 + b.*Pmax + c)./(d.*Pmax.^2 + e.*Pmax+f);
%==========================================================================
%========================PSO parameters==================================
nVar = n + 1; % 6 Generator output + 1 tie-line variable T
% Tie-line limits
%==========Tie line limits=============================================
Tmin = -100;
Tmax = 100;
%========================================================================
%========================Search bounds limits=============================
lb = [Pmin, Tmin];
ub = [Pmax, Tmax];
%==========================================================================
%============================PSO initialization===========================
swarmSize = 80;
maxIter = 600;
w = 0.72;% Initial weight
c1 = 1.49; c2 = 1.49;%
velMax = 0.2*(ub-lb); velMin = -velMax;
% ===========penalty coeefficient for power balance violations===========
penalty_factor = 1e6;
%===========initialization of PSO variable and preallocation=============
X = repmat(lb,swarmSize,1) + rand(swarmSize,nVar) .*(repmat(ub-lb,
swarmSize,1));
V = zeros(swarmSize,nVar);
pBest = X;
pBestVal = inf(swarmSize,1);
gBest = zeros(1,nVar);
gBestVal = inf;
convergence = zeros(maxIter,1);

%% ============================PSO MAIN LOOP============================
rng('default');
for iter = 1:maxIter
    for i = 1:swarmSize
        % ====================Max/Min bounds for P and T===================
        X(i,:) = max(X(i,:),lb);
        X(i,:) = min(X(i,:),ub);
        %==============================================================
```

```
        %================================Evaluate the function ===========
        [val,~] =
objective_funPSO(X(i,:),n,Pmin,Pmax,Tmin,Tmax,PD,Pwind,a,b,c,d,e,f,h,B_area
,penalty_factor);
        %================ update pBest=====================================
        if val < pBestVal(i)
            pBestVal(i) = val;
            pBest(i,:) = X(i,:);
        end
        %=====================update global best==========================
        if val < gBestVal
            gBestVal = val;
            gBest = X(i,:);
       end
    end

    % ===============Now update velocities and positions==================
    for i = 1:swarmSize
        r1 = rand(1,nVar); r2 = rand(1,nVar);
              V(i,:) = w*V(i,:) + c1.*r1.*(pBest(i,:) -X(i,:)) +
      c2.*r2.*(gBest - X(i,:));
        %==============Constraints for velocity===========================
        V(i,:) = max(min(V(i,:), velMax), velMin);
        X(i,:) = X(i,:) + V(i,:);
        %========================Check bounds for min/max==================
        X(i,:) = max(X(i,:),lb);
        X(i,:) = min(X(i,:), ub);
    end
    % ===================store convergence================================
convergence(iter) = gBestVal;
%======================================================================
if mod(iter,50)==0 || iter==1
    fprintf('Iter %d  best CEED+penalty = %.4f\n',iter, gBestVal);
end
  end
  %==========================global and Personal best=====================
  best = gBest;
  Pbest = best(1:n);
  Tbest = best(n+1);
  [~,details]=
  objective_funPSO(best,n,Pmin,Pmax,Tmin,Tmax,PD,Pwind,a,b,c,d,e,f,h,B_area,p
  enalty_factor);
  %% =======Fuel cost,emissioncosts,ceed_per_gen,fuelcost_total===========
  fuelcosts = a.*(Pbest.^2) + b.*Pbest + c;
  emissioncosts = d.*(Pbest.^2) + e.*Pbest + f;
  ceed_per_gen = fuelcosts + h.*emissioncosts;
  fuelcost_total = sum(fuelcosts);
  emission_total = sum(emissioncosts);
  ceed_total = sum(ceed_per_gen);
  % ====================Tramsmission losses===============================
  try
      B_area11 = permute(reshape(B_area,3,2,[]),[1 3 2]);
      P_resh = reshape(Pbest,3,1,[]);
      PL_area = bsxfun(@times,P_resh,reshape(P_resh,1,3,[])).*B_area11;
      PLvec = sum(reshape(PL_area,[],2));
  catch
      PLvec = [0.001, 0.001];
  end
  P1 = Pbest(1:3); P2 = Pbest(4:6);
  sumP1 = sum(P1); sumP2 = sum(P2);

  fprintf('\n=====PSO results summary=====\n');
```

```
fprintf('Iterations performed: %d\n',maxIter);
fprintf('Best tie-line (T) = %.4f MW (positive: area1 -> area2\n',Tbest);
fprintf('Generator dispatch (P1..P6) [MW]:\n'); disp(Pbest');
fprintf('Area1 generation sum = %.4f MW, Area2 generation sum = %.4f MW\n',
sumP1,sumP2);
fprintf('Expected Wind per area (MW):\n'); disp(Pwind');
fprintf('Transmission losses per area (MW):'); disp(PLvec);
fprintf('Total Fuel cost = %.4f\n',fuelcost_total);
fprintf('Total Emission cost = %.4f\n',emission_total);
fprintf('Total CEED (fuel + h*emission) = %.4f\n', ceed_total);
fprintf('Power balance residuals area1, area2: [res1 res2] = [%.6f
%.6f\n',details.residuals);
toc
%===========================Convergence output plot========================
figure
plot(1:maxIter,convergence, 'lineWidth', 1.5);
xlabel('Iteration'); ylabel('Best objective (CEED + penalties)');
title('PSO Convergence');
grid on;

 finalvalue = [Pbest sum(Pwind(:,:)) sum(PLvec(:,:)) fuelcost_total
emission_total ceed_total];
 finalvalue'
 digits(6)
vpa(finalvalue')
%=========This is the objective function for the main program:
MAWTEED_casePSO6units========
function [objVal, details] =
objective_funPSO(X,n,Pmin,Pmax,Tmin,Tmax,PD,Pwind,a,b,c,d,e,f,h,B_area,pena
lty_factor)
Pgen = X(1:n); % Generator outputs
T = X(n+1);% Tie line flow
%===================Generator constraints=================================
Pgen = max(Pgen,Pmin);
Pgen = min(Pgen,Pmax);
T = max(min(T,Tmax),Tmin);
%=========================================================================
%===================split generators to two areas=========================
P1 = Pgen(1:3);
P2 = Pgen(4:6);
%=========================================================================
%=====================Fuel cost and emission==============================
fuelcosts = a.*(Pgen.^2) + b.*Pgen + c;
emission = d.*(Pgen.^2) + e.*Pgen + f;
ceed_per_gen = fuelcosts+h.*emission;
fuel_total = sum(fuelcosts);
emission_total = sum(emission);
ceed_total = sum(ceed_per_gen);
% ====================Tramsmission losses=================================
try
    B_area11 = permute(reshape(B_area,3,2,[]),[1 3 2]);
    P_resh = reshape(Pbest,3,1,[]);
    PL_area = bsxfun(@times,P_resh,reshape(P_resh,1,3,[])).*B_area11;
    PLvec = sum(reshape(PL_area,[],2));
catch
    PLvec = [0.001, 0.001];
end
%=========================================================================
%====================Power Balance per area===============================
%========Area1 demand with wind and tie line export=======================
balance1 = sum(P1) + Pwind(1)-PD(1)-PLvec(1)-T;
%========Area2 demand with wind and tie line import=======================
```

```
balance2 = sum(P2) + Pwind(2)-PD(2)-PLvec(2)+T;
residuals = [balance1, balance2];
%===================Calcualtion of penalty================================
penalty = penalty_factor*(residuals(1)^2 + residuals(2)^2);
%=================Final objective value===================================
objVal = ceed_total + penalty;
%=========================================================================
%=================Values under varible name: details=====================
details.fuel_total = fuel_total;
details.emission_total = emission_total;
details.ceed_total = ceed_total;
details.PLvec = PLvec;
details.residuals=residuals;
details.Pgen = Pgen;
details.T = T;
%=========================================================================
end
```

**APPENDIX D2:** MATLAB script file – MAWTEED_casePSO12units.m

```
%=========================================================================
%Main program name: MAWTEED_casePSO12units
%============WTEED problem with 12 generators and 4 areas, 3 generators
%in each areas with wind power 330MW. This program isused to create table
%6.3, and used in table 6.4 as well. It is used with the function
%objective_funPSO_4area.m
clearvars;clc;close all;tic;
%=================Problem Data============================================
m=4; % No of areas
n=12; % Total No of Generators in all the area
NG=3; % No of Generators in each area
TL=6; % NO of Tie lines
PD=[499.9280 410.0580 579.9750 600.0390]; % Area power demand

%====================Generator limits=====================================
Pmin = [35 130 125 10 35 125 15 30 50 15 30 50];
Pmax = [210 325 315 150 110 215 175 215 335 175 215 335];
%=========================================================================
%%============Multi area Generator fuel Cost coefficients==============
a={[0.03546 0.02111 0.01799],[0.15247 0.02803 0.14834],...
   [0.10587 0.07505 0.11934],[0.10587 0.13552 0.08963]};
b={[38.30553 36.32782 38.27041],[38.53973 40.39655 38.34001],...
   [46.15916 43.83562 50.63211],[46.15916 41.03782 33.56211]};
c={[1243.5311 1658.5696 1356.6592],[0756.7989 0449.9977 0558.5696],...
   [0451.3251 0673.0267 0530.7199],[0851.3251 1038.533 1285.907]};
%%============Multi area Emission coefficients========================
d={[0.00683 0.00461 0.00461],[0.00484 0.00754 0.00661],...
   [0.00914 0.00533 0.00674],[0.00728 0.00479 0.00387]};
e={[-0.54551 -0.51160 -0.51160],[-0.32767 -0.54551 -0.63262],...
   [-0.43211 -0.61173 -0.49731],[-0.6821 -0.50660 -0.49340]};

f={[40.26690 42.89553 42.89553],[33.85932 50.639310 45.83267],...
   [48.21560 52.45210 41.10420],[30.36320 25.17650 27.75490]};
%============Tie line limits==============================================
PTmin_mj=[5 5 5 5 5 5];
PTmax_mj=[60 50 60 60 60 50];
%=========================================================================
B_area=[ 0.000071 0.00003 0.000025
         0.00003 0.000069 0.000032
         0.000025 0.000032 0.00008
         0.000056 0.000045 0.000015
```

```
            0.000023 0.000042 0.000047
            0.000032 0.000023 0.000027
            0.00002 0.000028 0.000053
            0.000086 0.000034 0.000016
            0.000053 0.000016 0.000028
            0.000074 0.00003 0.000025
            0.000049 0.000069 0.000037
            0.000022 0.000032 0.000083];
    B_area1={B_area((1:3),:),B_area((4:6),:),B_area((7:9),:),B_area((10:12),:)}
    ;
    %==============================Max/Max penalty factor h======================
    h = cell(1,m);
    for i = 1:m
        h{i} = (a{i}.*Pmax((i-1)*NG+(1:NG)).^2 + b{i}.*Pmax((i-1)*NG+(1:NG)) +
    c{i})./...
               (d{i}.*Pmax((i-1)*NG+(1:NG)).^2 + e{i}.*Pmax((i-1)*NG+(1:NG)) +
    f{i});
    end
    % ====================Wind power data for each area=================
    windData = {[9.891 10.785 6.861 14.482 12.184 4.602];...
                [8.5 7.2 6.9 10.4 12.1 5.8];...
                [7.3 8.6 6.2 11.0 9.8 4.5];...
                [6.5 7.9 5.1 9.2 8.7 4.0];};
    Prated = [215 210 210 210]; %Rated wind power per area in (MW)
    Vin = 5; Vr = 15; Vout = 25;
    vbins = 0:0.5:30;
    Ns = 50; %Number of scenarios
    TotalwindPower = zeros(m,Ns);
    for area = 1:m
        v0 = windData{area}';
        pwind = fitdist(v0,'Weibull');
        %=============== Power for each area=================================
         pwvbins = zeros(size(vbins));
   pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
   ((vbins(vbins>=Vin &vbins<Vr).^2 - Vin^2) /(Vr^2 - Vin^2));
   pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
% ================Expected wind power=================================
   pdfv = pdf(pwind,vbins);
   avgpower = trapz(vbins,pdfv .*pwvbins);
  fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
  for s =1:Ns
      v = random(pwind);
      if v<Vin
      Pw = 0;
      elseif v < Vr
      Pw = Prated(area) * ((v^2 - Vin^2)/(Vr^2 - Vin^2));
      elseif v < Vout
      Pw = Prated(area);
      else
      Pw = 0;
      end
    TotalwindPower(area,s) = Pw;
  end
    end
    %====================Expected wind power ==============================
    Pwind_exp = mean(TotalwindPower,2);
    Pwind = Pwind_exp; % Expected wind power

    %% ======================PSO Parameters===================================
    nVar = n +TL; % 12 generators + 6 tie-lines
    lb = [Pmin,PTmin_mj]; ub = [Pmax,PTmax_mj];
    swarmSize = 100; maxIter = 500; w = 0.9; c1 = 2; c2 = 2;
```

```
Vmax = 0.2*(ub-lb); Vmin = -Vmax;
penalty_factor = 1e6;
%===========initialization of PSO variable and preallocation=============
X = repmat(lb,swarmSize,1) + rand(swarmSize,nVar) .*(repmat(ub-lb,
swarmSize,1));
V = zeros(swarmSize,nVar);
pBest = X;
pBestVal = inf(swarmSize,1);
gBest = zeros(1,nVar);
gBestVal = inf;
convergence = zeros(maxIter,1);
%%==============================PSO main loop===========================
rng('default');
for iter = 1:maxIter
    for i =1:swarmSize
        X(i,:) = max(X(i,:),lb);
       X(i,:) = min(X(i,:),ub);
%================================Evaluate the function ==================
        [val, ~] =
objective_funPSO_4area(X(i,:),n,m,NG,TL,Pmin,Pmax,PTmin_mj,PTmax_mj,PD,Pwin
d,a,b,c,d,e,f,h,B_area1,penalty_factor);
%================== update pBest=========================================
        if val < pBestVal(i)
            pBestVal(i) = val;
            pBest(i,:) = X(i,:);
        end
%====================update global best==================================
        if val < gBestVal
            gBestVal = val;
            gBest = X(i,:);
       end
    end
 % ===============Now update velocities and positions=================
    for i = 1:swarmSize
        r1 = rand(1,nVar); r2 = rand(1,nVar);
        V(i,:) = w*V(i,:) + c1.*r1.*(pBest(i,:) -X(i,:)) + c2.*r2.*(gBest -
X(i,:));
 %======================Constraints for
velocity============================
        V(i,:) = max(min(V(i,:), Vmax), Vmin);
        X(i,:) = X(i,:) + V(i,:);
 %===============================Check bounds for min/max==================
        X(i,:) = max(X(i,:),lb);
        X(i,:) = min(X(i,:), ub);
    end
    % ==================store convergence================================
    convergence(iter) = gBestVal;
    %==================================================================
    if mod(iter,50)==0 || iter==1
        fprintf('Iter %d  best CEED+penalty = %.4f\n',iter, gBestVal);
    end
end
%%======================Get the best sulution============================
[~, details] =
objective_funPSO_4area(gBest',n,m,NG,TL,Pmin,Pmax,PTmin_mj,PTmax_mj,PD,Pwin
d,a,b,c,d,e,f,h,B_area1,penalty_factor);
fprintf('\n=====PSO Results Summary====\n');
fprintf('Generator outputs [MW]:\n'); disp(details.Pgen);
fprintf('Tie-line flows[MW]:\n'); disp(details.Tlines);
fprintf('Expected Wind per area (MW):\n'); disp(Pwind');
fprintf('Transmission losses per area (MW):\n'); disp(details.PL_area);
fprintf('Total Fuel cost = %.4f\n',details.fuel_total);
```

```
fprintf('Total Emission cost = %.4f\n',details.emission_total);
fprintf('Total CEED (fuel + h*emission) = %.4f\n', details.ceed_total);
fprintf('Losses = %.4f\n',details.PL_area);
% ===========================Convergence output
plot========================
figure
plot(1:maxIter,convergence, 'lineWidth', 1.5);
xlabel('Iteration'); ylabel('Best objective (CEED + penalties)');
title('PSO Convergence');
grid on;
% finalvalue = [gBest' sum(Pwind(:,:)) sum(PL_area(:,:)) fuel_total
emission_total ceed_total];
%  finalvalue'
%  digits(6)
% vpa(finalvalue')
toc
Totalwind = sum(Pwind(:,:))

%=========This is the objective function named: objective_funPSO_4area.m
%used with the main program: MAWTEED_casePSO12units
function [objVal, details] =
objective_funPSO_4area(X,n,m,NG,TL,Pmin,Pmax,PTmin_mj,PTmax_mj,PD,Pwind,a,b
,c,d,e,f,h,B_area1,penalty_factor)

%X =[P1..P12, T1..T6]
Pgen = X(1:n); % Generator outputs
Tlines = X(n+1:end);% Tie line flow
%===================Generator constraints=================================
Pgen = max(Pgen,Pmin);
Pgen = min(Pgen,Pmax);
Tlines = max(min(Tlines,PTmax_mj),PTmin_mj);
%%====================Split generator outputs by area=====================
P_area = cell(1,m);
% for i=1:m
for i =1:m
    idx = (i-1)*NG + (1:NG);
    P_area{i} = Pgen(idx);
end
%%========================Transmission losses=============================
PL_area = zeros(1,m);
for i=1:m
    P_resh = P_area{i}(:);
    PL_matrix = (P_resh*P_resh').*B_area1{i};
    PL_area(i) = sum(PL_matrix(:));
end
%%===================Tie-line distribution to areas======================
T_area = zeros(1,m);
%================distribution sum of tie lines==========================
T_area(1) = sum(Tlines(1:TL/2)); % area1 receives or sends
T_area(2) = sum(Tlines(1:TL/2));% Area2
T_area(3) =  sum(Tlines(TL/2+1:end)); %area3
T_area(4) =  sum(Tlines(TL/2+1:end)); %area4

%%============= Calculation of the CEED problem==========================
ceed_total=0;
fuel_total =0;
emission_total =0;
for i=1:m
    fuel = a{i}.*P_area{i}.^2 + b{i}.*P_area{i} + c{i};
    emission = d{i}.*P_area{i}.^2 + e{i}.*P_area{i} + f{i};
    ceed_per_gen = fuel + h{i}.*emission;
    fuel_total = fuel_total+ sum(fuel);
```

```
    emission_total = emission_total + sum(emission);
    ceed_total = ceed_total + sum(ceed_per_gen);
end
%%==========================Power balance equation========================
residuals = zeros(1,m);
for i=1:m
    residuals(i) = sum(P_area{i}) + Pwind(i) -PD(i)-PL_area(i)+T_area(i);
end
%%====================Penalty for power balance============================
penalty = penalty_factor*sum(residuals.^2);
%%====================Final objective======================================
objVal = ceed_total + penalty;
%% ====================Variables under details===========================
details.Pgen = Pgen;
details.Tlines = Tlines;
details.P_area = P_area;
details.PL_area = PL_area;
details.residuals = residuals;
details.fuel_total = fuel_total;
details.emission_total = emission_total;
details.ceed_total = ceed_total;
end
```

**APPENDIX D3:** MATLAB script file – MAWTEED_casePSO40units.m

```
% Main program name: MAWTEED_casePSO40units. Program used in generating
% table 6.7 and also used in 6.8. Used with the function
named:objective_funPSO_2area
%%======================2-Area 40 Generator MAWTEED problem using
PSO=============
clearvars; clc; close all; tic;
%%==========================Problem Data================================
m = 2; n = 40; TL =1;% arears, generators, and tie lines
PD = [7500 3000];% Power demand for both areas
%% ============Data are given in the reference paper (Basu, 2013)==========
%The data are given in the format[GenNum Pmin Pmax a b c d e f]============
Data=[.....
1 36 114 0.0069  6.73 94.705 0.048  -2.22  60
2 36 114 0.0069  6.73 94.705 0.048  -2.22  60
3 60 120 0.02028 7.07 309.54 0.0762 -2.36  100
4 80 190 0.00942 8.18 369.03 0.054  -3.14  120
5 47 97  0.0114  5.35 148.89 0.085  -1.89  50
6 68 140 0.01142 8.05 222.33 0.0854 -3.08  80
7 110 300 0.00357 8.03 287.71 0.0242 -3.06 100
8 135 300 0.00492 6.99 391.98 0.031  -2.32 130
9 135 300 0.00573 6.6 455.76 0.0335 -2.11 150
10 130 300 0.00605 12.9 722.82 0.425 -4.34 280
11 94 375 0.00515 12.9 635.2 0.0322 -4.34 220
12 94 375 0.00569 12.8 654.69 0.0338 -4.28 225
13 125 500 0.00421 12.5 913.4 0.0296 -4.18 300
14 125 500 0.00752 8.84 1760.4 0.0512 -3.34 520
15 125 500 0.00752 8.84 1760.4 0.0496 -3.55 510
16 125 500 0.00752 8.84 1760.4 0.0496 -3.55 510
17 220 500 0.00313 7.97 647.85 0.0151 -2.68 220
18 220 500 0.00313 7.95 649.69 0.0151 -2.66 222
19 242 550 0.00313 7.97 647.83 0.0151 -2.68 220
20 242 550 0.00313 7.97 647.81 0.0151 -2.68 220
21 254 550 0.00298 6.63 785.96 0.0145 -2.22 290
22 254 550 0.00298 6.63 785.96 0.0145 -2.22 285
23 254 550 0.00284 6.66 794.53 0.0138 -2.26 295
24 254 550 0.00284 6.66 794.53 0.0138 -2.26 295
```

```
25 254 550 0.00277 7.1 801.32 0.0132 -2.42  310
26 254 550 0.00277 7.1 801.32 0.0132 -2.42  310
27 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
28 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
29 10 150 0.52124 3.33 1055.1 1.842 -1.11   360
30 47 97  0.0114  5.35 148.89 0.085 -1.89   50
31 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
32 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
33 60 190 0.0016 6.43 222.92  0.0121 -2.08   80
34 90 200 0.0001 8.95 107.87  0.0012 -3.48   65
35 90 200 0.0001 8.62 116.58  0.0012 -3.24   70
36 90 200 0.0001 8.62 116.58  0.0012 -3.24   70
37 25 110 0.0161 5.88 307.45  0.095  -1.98   100
38 25 110 0.0161 5.88 307.45  0.095  -1.98   100
39 25 110 0.0161 5.88 307.45  0.095  -1.98   100
40 242 550 0.00313 7.97 647.83 0.0151 -2.68  220];
%% ======================Multi area fuel cost coefficients================
a=Data(:,4)'; a={a(1:20),a(21:40)};
b=Data(:,5)'; b={b(1:20),b(21:40)};
c=Data(:,6)'; c={c(1:20),c(21:40)};
%%=====================Multi area Emmission coefficients===================
d=Data(:,7)'; d={d(1:20),d(21:40)};
e=Data(:,8)'; e={e(1:20),e(21:40)};
f=Data(:,9)'; f={f(1:20),f(21:40)};
%% ===============Multi area Generator real power limits====================
Pmin=Data(:,2)';Pmin1={Pmin(1:20),Pmin(21:40)};
Pmax=Data(:,3)';Pmax1={Pmax(1:20),Pmax(21:40)};
%=========================================================================
% % Tie-line limits and coefficients
%==============Tie-line limits and coefficients============================
PTmin_mj = 100; PTmax_mj = 1500;
% PTmax_jm = 100; PTmax_jm = 1500;
% q_mj = rand; q_jm = rand;
% flr_jm = rand;
% alpha_mj = 0.01; alpha_jm = 0.01;
% ====================Wind power data for each area=================
windData = {[9.891 10.785 6.861 14.482 12.184 4.602],[8.5 7.2 6.9 10.4 12.1
5.8]};
Prated = [220*3 0];
Vin = 3; Vr = 12; Vout = 25;
vbins = 0:0.5:30;
Ns = 50; %Number of scenarios
TotalwindPower = zeros(m,Ns);
Pwind_exp = zeros(m,1);
Pwind_limit = 330;% Wind power limits for system wind of 330MW
for area = 1:m
    v0 = windData{area}';
    pwind = fitdist(v0,'Weibull');
    %=============== Power for each area===================================
     pwvbins = zeros(size(vbins));
     pwvbins(vbins>=Vin & vbins<Vr)= Prated(area).*...
     ((vbins(vbins>=Vin &vbins<Vr).^2 - Vin^2) /(Vr^2 - Vin^2));
     pwvbins(vbins>=Vr & vbins<=Vout ) = Prated(area);
  % ================Expected wind power====================================
     pdfv = pdf(pwind,vbins);
     avgpower = trapz(vbins,pdfv .*pwvbins);
     Pwind_exp(area) = avgpower;
    fprintf('Area %d: Expected Wind ≈ %.2f MW\n',area,avgpower);
    for s =1:Ns
        v = random(pwind);
        if v<Vin
        Pw = 0;
```

```
        elseif v < Vr
        Pw = Prated(area) * ((v^2 - Vin^2)/(Vr^2 - Vin^2));
        elseif v < Vout
        Pw = Prated(area);
        else
        Pw = 0;
        end
  TotalwindPower(area,s) = Pw;
    end
end
%============================Make sure the wind power doesnt reach 330MW==
totalExp = sum(Pwind_exp);
if totalExp > Pwind_limit
    scaleFactor = Pwind_limit/totalExp;
    Pwind_exp = Pwind_exp*scaleFactor;
    TotalwindPower = TotalwindPower*scaleFactor;
fprintf('Whole wind limit: of the dispatch problem (%.2f)\n',scaleFactor);
end
%==================The expected wind power is =============================
Pwind = Pwind_exp;
fprintf('Total expected wind after limit check: %.2f MW\n',sum(Pwind));
%% ======================PSO Parameters===================================
nVar = n +TL; % 12 generators + 6 tie-lines
lb = [Pmin,PTmin_mj]; ub = [Pmax,PTmax_mj];
swarmSize = 100; maxIter = 1000; w = 0.9; c1 = 2; c2 = 2;
Vmax = 0.2*(ub-lb); Vmin = -Vmax;
penalty_factor = 1000000;
%============initialization of PSO variable and preallocation==============
X = repmat(lb,swarmSize,1) + rand(swarmSize,nVar) .*(repmat(ub-lb,
swarmSize,1));
V = zeros(swarmSize,nVar);
pBest = X;
pBestVal = inf(swarmSize,1);
gBest = zeros(1,nVar);
gBestVal = inf;
convergence = zeros(maxIter,1);
%%==============================PSO main loop============================
rng('default');
for iter = 1:maxIter
    for i =1:swarmSize
        X(i,:) = max(X(i,:),lb);
        X(i,:) = min(X(i,:),ub);
        % [val, details] = objective_funPSO_2area(X(i,:), n, m, TL, PD,
Pwind, ...
        %     a,b,c,d,e,f,Pmin,Pmax, Pmin1,Pmax1,
q_mj,q_jm,alpha_mj,alpha_jm, flr_jm, penalty_factor);
[val, details] = objective_funPSO_2area(X(i,:), n, m, TL, PD, Pwind, ...

a,b,c,d,e,f,Pmin,Pmax,Pmin1,Pmax1,penalty_factor,PTmin_mj,PTmax_mj);
%================== update pBest==========================================
        if val < pBestVal(i)
            pBestVal(i) = val;
            pBest(i,:) = X(i,:);
        end
%=====================update global best==================================
        if val < gBestVal
            gBestVal = val;
            gBest = X(i,:);
            gBestDetails = details;
        end
    end
 % ===============Now update velocities and positions=================
```

```
    for i = 1:swarmSize
        r1 = rand(1,nVar); r2 = rand(1,nVar);
        V(i,:) = w*V(i,:) + c1.*r1.*(pBest(i,:) -X(i,:)) + c2.*r2.*(gBest -
X(i,:));
 %=======================Constraints for
velocity============================
        V(i,:) = max(min(V(i,:), Vmax), Vmin);
        X(i,:) = X(i,:) + V(i,:);
 %===============================Check bounds for min/max==================
        X(i,:) = max(X(i,:),lb);
        X(i,:) = min(X(i,:), ub);
    end
%========================store convergence=================================
    convergence(iter) = gBestVal;
%==========================================================================
    if mod(iter,50)==0 || iter==1
        fprintf('Iter %d  best CEED+penalty = %.4f\n',iter, gBestVal);
    end
end
% [~, details] = objective_funPSO_2area(gBest', n, m, TL, PD, Pwind, ...
%           a,b,c,d,e,f,Pmin,Pmax, Pmin1,Pmax1, q_mj,q_jm,alpha_mj,alpha_jm,
flr_jm, penalty_factor);
[~, details] = objective_funPSO_2area(gBest', n, m, TL, PD, Pwind, ...
         a,b,c,d,e,f,Pmin,Pmax, Pmin1,Pmax1,
penalty_factor,PTmin_mj,PTmax_mj);
%%===========================WTEED best solution===========================
disp('======PSO WTEED Results=======');
fprintf('Total Fuel cost = %.4f\n',gBestDetails.fuel_total);
fprintf('Total Emission cost = %.4f\n',gBestDetails.emission_total);
fprintf('Total CEED (fuel + h*emission) = %.4f\n',gBestDetails.ceed_total);
fprintf('Generator outputs per area [MW]:\n'); disp(gBestDetails.Pgen);
fprintf('Tie-line flows [MW]:\n'); disp(gBestDetails.Tlines);
fprintf('Expected Wind per area [MW]:\n'); disp(Pwind');
%===========================Convergence output plot========================
figure
plot(1:maxIter,convergence, 'lineWidth', 1.5);
xlabel('Iteration'); ylabel('Best objective (CEED + penalties)');
title('PSO Convergence');
grid on;
%==========================================================================

%=========This is the objective function named: objective_funPSO_2area.m
%used with the main program: MAWTEED_casePSO40units

function [objVal, details] = objective_funPSO_2area(X, n, m, TL, PD, Pwind,
...
    a,b,c,d,e,f,Pmin,Pmax, Pmin1,Pmax1, penalty_factor,PTmin_mj,PTmax_mj)
%=================Objective function=======================================
Pgen = X(1:n);
T = X(n+1:end);% Tie line flows TL =1;
%==================Generator constraints===================================
Pgen = max(Pgen,Pmin);
Pgen = min(Pgen,Pmax);
T = max(min(T,PTmax_mj),PTmin_mj);
%=========================Split generator per area=========================
P1 = Pgen(1:20);
P2 = Pgen(21:40);
%======================Fuel cost and Emission==============================
fuelcosts = [a{1}.*P1.^2 + b{1}.*P1 + c{1},a{2}.*P2.^2 + b{2}.*P2 + c{2}];
emission = [d{1}.*P1.^2 + e{1}.*P1 + f{1},d{2}.*P2.^2 + e{2}.*P2 + f{2}];
%==========================Penalty factor for the two areas==============
```

```
h1 = (a{1}.*Pmax1{1}.^2 + b{1}.*Pmax1{1} + c{1})./ (d{1}.*Pmax1{1}.^2 +
e{1}.*Pmax1{1} + f{1});
h2 = (a{2}.*Pmax1{2}.^2 + b{2}.*Pmax1{2} + c{2})./ (d{2}.*Pmax1{2}.^2 +
e{2}.*Pmax1{2} + f{2});
h = [h1 h2];
ceed_per_gen = fuelcosts + h.*emission;
fuel_total = sum(fuelcosts);
emission_total = sum(emission);
ceed_total = sum(ceed_per_gen);
%=====================Power balance residuals==============================
balance1 = sum(P1)+ Pwind(1)+(-T)-PD(1); % Area 1 tie line export
balance2 = sum(P2)+ Pwind(2)+ (T)-PD(2); % Area 2 import tie line

residuals = [balance1, balance2];
%==========================Penalty=========================================
penalty = penalty_factor* sum(residuals.^2);
%=======================Objective value====================================
objVal = ceed_total + penalty;

%======================Name under variable details=========================
details.fuel_total = fuel_total;
details.emission_total = emission_total;
details.ceed_total = ceed_total;
details.residuals = residuals;
details.Pgen = {P1, P2};
details.Tlines = T;
end.
%==========================================================================
```