OPTIMAL ECONOMIC AND ENVIRONMENTAL OPERATION OF ELECTRIC POWER SYSTEMS VIA MODERN META-HEURISTIC OPTIMIZATION ALGORITHMS

by

Fahad Saad Abu-Mouti

Submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
December 2011

DALHOUSIE UNIVERSITY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "OPTIMAL ECONOMIC AND ENVIRONMENTAL OPERATION OF ELECTRIC POWER SYSTEMS VIA MODERN META-HEURISTIC OPTIMIZATION ALGORITHMS" by Fahad Saad Abu-Mouti in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Dated: December 06, 2011

External Examiner: ________________________________

Research Supervisor: ________________________________

Examining Committee: ________________________________

____

Departmental Representative: ____________________________

DALHOUSIE UNIVERSITY

DATE: December 06, 2011

AUTHOR: Fahad Saad Abu-Mouti

TITLE: OPTIMAL ECONOMIC AND ENVIRONMENTAL OPERATION OF ELECTRIC POWER SYSTEMS VIA MODERN META-HEURISTIC OPTIMIZATION ALGORITHMS

DEPARTMENT OR SCHOOL: Department of Electrical and Computer Engineering

DEGREE: Ph.D. CONVOCATION: May YEAR: 2012

Signature of Author

DEDICATION PAGE

To my beloved parents.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Due to world-wide escalating fuel costs, increasing demand for electricity, and growing concern for the environment, power utilities strive for optimal economic operation of their electric networks. Striking a balance between profitable energy choices and environmentally-friendly practices is the main goal of this thesis. The dynamic economic dispatch (DED) occupies a prominent place in power system operation and control. However, there is a paucity of studies of the DED problem, as it has not been as thoroughly investigated as other electric power system optimization areas. The nonlinear and non-convex characteristics are more prevalent in the DED problem. Therefore, it is possible that computational methods may not yield a global extrema as many local extrema may be encountered, and, in this case, obtaining a truly optimal solution presents a challenge.

Two modern meta-heuristic optimization algorithms are utilized to solve the DED problem. The artificial bee colony (ABC) algorithm is a recently introduced population-based algorithm motivated by the intelligent foraging behaviour of the honeybee swarm. This thesis proposes a novel meta-heuristic optimization algorithm inspired by the intelligent behaviour or survival instincts of a sensory-deprived human being. The sensory-deprived optimization algorithm (SDOA) uses the exploration and exploitation processes simultaneously and distinctly from other algorithms. After solving different benchmark optimization functions, the SDOA efficiency is evident with results that outperform or match those attained by other well-known methods.

To enhance the utilized algorithms' performances in solving the dynamic economic and emission dispatch problems, a new constrained search-tactic is offered. Two groups of test cases are used to validate the effectiveness of the proposed algorithms. The first one is designated to solve the single objective function scenario and to verify the presented constrained search-tactic. The second group focuses on the multiple objective functions' scenario as well as an attempt to integrate a renewable source and analyze its impact. The outcomes of ABC and SDOA algorithms are compared with those of other older and known methods. The promising results in both utilized algorithms show great potential that can be employed in several electric power system optimization areas.

LIST OF ABBREVIATIONS AND SYMBOLS USED

a, b, c, d, g	Fuel Cost Function Coefficients
ABC	Artificial Bee Colony
ABCP	Artificial Bee Colony Programming
ACO	Ant Colony Optimization
AIS	Artificial Immune System
ANN	Artificial Neural Networks
B	Coefficient Loss Matrix
BF	Bacterial Foraging
BP	Back Propagation
CABC	Chaotic Artificial Bee Colony
CDE	Chaotic Differential Evolution
CMOS	Complementary Metal Oxide Semiconductor
CO_2	Carbon Dioxide
CPU	Central Processing Unit
CS	Colony Size
D	Dimensions of a Problem
DABC	Discrete Artificial Bee Colony
DE	Differential Evolutionary
DED	Dynamic Economic Dispatch
DEED	Dynamic Economic Emission Dispatch
DG	Distributed Generation

DP	Dynamic Programming
DR	Lower Limit of the Generation Unit's Ramp-Rate
e	Exponential
E	Emission Objective Function
E_b	Employ Bee
ED	Economic Dispatch
EP	Evolutionary Programming
ES	Evolution Strategy
ESO	Evolution Strategy Optimization
F	Fuel Objective Function
FA	Firefly Algorithm
FEP	Fast Evolutionary Programming
fitness or *fit*	Fitness Value of a Solution
FRAH	Feasibility Restoration Attempt Heuristic
GA	Genetic Algorithm
GENCO	Generation Company
GHG	Green House Gas
GW	Giga Watt
GWEC	Global Wind Energy Council
h	Hour
HDE	Hybrid Differential Evolution
HHS	Hybrid Harmony Search
HNN	Hopfield Neural Networks
HS	Harmony Search

HTS	Hydro-Thermal Scheduling
ICH	Incremental Cost based Optimization Heuristic
IDE	Improved Differential Evolutionary
IHS	Improved Harmony Search
IIR	Infinite Impulse Response
IP	Interior Point
lb	Pound, Weight Unit
LCMST	Leaf-Constrained Minimum Spanning Tree
limit	Number of Trials to Generate a Scout Bee
LP	Linear Programming
LRS	Local Random Search
LSQ	Least Square Error
MABC	Modified Artificial Bee Colony
max	Maximum Value
MCN	Maximum Cycle Number
MESFET	Metal Extended Semiconductor Field Effect Transistor
min	Minimum Value
MPC	Model Predictive Control
MR	Modification Rate
MTS	Multiple Tabu Search
MW	Mega Watt
N	Number of Committed Generation Units
n	Number of Prohibited Operating Zones of Committed Generation Units
NLP	Nonlinear Programming

NN	Neural Networks
NO_x	Nitrogen Oxides
NSGA-II	Non-Dominated Sorting Genetic Algorithm-II
O_b	Onlooker Bee
OCDD	Optimal Control Dynamic Dispatch
OPF	Optimal Power Flow
p	Probability of a Solution
P	Real Generated Output Power
PAR	Pitch Adjustment Rate
PC	Personal Computer
P_D	Total System Real Power Demand
PFSP	Permutation Flow Shop Scheduling Problem
PI	Proportional Integral
PID	Proportional Integral Derivative
P^l	Lower Bound on the Prohibited Operating Zone
P_L	Overall System Real Power Losses
P_{max}	Maximum Real Power Output Limit of a Generation Unit
P_{min}	Minimum Real Power Output Limit of a Generation Unit
PNSGA	Preference-based Non-Dominated Sorting Genetic Algorithm
PS	Pattern Search
PSO	Particle Swarm Optimization
P^u	Upper Bound on the Prohibited Operating Zone
PV	Photovoltaic
QEA	Quantum Evolutionary Algorithm

QP	Quadratic Programming
R	Moving Steps of a Sensory-Deprived Person
rad	Radian
RAM	Random Access Memory
RC-GA	Real Coded Genetic Algorithm
RNN	Recurrent Neural Networks
RS	Renewable Source
s	Second
SA	Simulating Annealing
SDOA	Sensory-Deprived Optimization Algorithm
SDP	Sensory-Deprived Person
$SDP_{parallel}$	Sensory-Deprived Person of the Parallel Search
SED	Static Economic Dispatch
SF	Scaling Factor
SO_2	Sulfur Dioxide
SOA	Seeker Optimization Algorithm
SQP	Sequential Quadratic Programming
Std.Dev.	Standard Deviation
T	Predetermined Dispatch Period
t	Time
TS	Tabu Search
u	Uniform Variant Multiplier
U	Uniform Variant Multiplier Excludes the *u* Interval
UC	Unit Commitment

UR	Upper Limit of the Generation Unit's Ramp-Rate
VSHDE	Variable Scaling Hybrid Differential Evolution
w	Weight Factor
w.r.t.	With Respect to
WRLP	Weighted Ring Loading Problem
WSN	Wireless Sensory Network
WT	Wavelet Transform
x	Possible Solution
x^*	Global Solution
α , β, γ, η, δ	Emission Function Coefficients
ζ	Price Associated with the Emissions
μ_{RS}	Multiplier Represents the Permissible Real Power by Renewable Source
φ	Predetermine Iteration's Number
$\boldsymbol{\Psi}$	Combined Economic and Emission Objective Functions

ACKNOWLEDGEMENTS

I would like to express great appreciation to my supervisor Dr. M. E. El-Hawary for his guidance and encouragement during my graduate studies. I have learned immeasurably from him, not only academically, but also about various aspects of life. It is an honor to be one of Dr. El-Hawary's graduate students.

A special thank you is extended to my advisory committee members, Dr. Timothy Little and Dr. William Phillips, for their valuable comments and insightful discussions during my graduate studies.

I would like to thank my parents, wife, daughter, brothers, and sister for their patience and endless support during my research. Your belief in me has always been an important motivation. I am especially grateful to my brother Omer for his unwavering and unforgettable support during my Ph. D. study.

Last, but not least, I would like to convey my gratitude to the Kingdom of Saudi Arabia, represented by the Ministry of Higher Education – Saudi Arabian Cultural Bureau in Canada – for sponsoring me through my graduate studies at Dalhousie University.

CHAPTER 1: INTRODUCTION

1.1 MOTIVATION

Power utilities strive for optimal economic operation of their electric networks while considering the challenges of escalating fuel costs and increasing demand for electricity. Their economic objectives suffer, however, when environmental constraints are considered. Striking a balance between profitable energy choices and environmentally-friendly practices is the main goal of current research in this field. The ultimate goal of power plants is to meet the required load demand with the lowest operating costs possible while taking into consideration practical equality and inequality constraints. At the same time, environmental concerns, such as gas emissions caused by fossil fuels, considerably affect this goal. Power utilities endeavor to minimize both fuel costs and emissions simultaneously.

Optimal operation of electric power system networks is a challenging real-world engineering problem. Indeed, the optimal operation of these networks is the result of multiple optimization problems that interact with each other sufficiently and efficiently. Those – linked – optimization problems are the unit commitment (UC), optimal power flow (OPF), static economic dispatch (SED), hydro-thermal scheduling (HTS), dynamic economic dispatch (DED). However, there is not as many studies of the DED problem – based on a review of the IEL and Science Direct databases – and it has not been as thoroughly investigated as other electric power system optimization areas, as is evident in Figure 1.1. The DED occupies a prominent place in a power system's operation and control. It aims to determine the optimal power outputs of on-line generating units in order to meet the load demand subject to satisfying various operational constraints over finite dispatch periods. Similar to most real-world complex engineering optimization problems, the nonlinear and non-convex characteristics are more prevalent in the DED problem. Therefore, it is possible that computational methods may not yield a global extremum as many local extrema may be encountered, and, in this case, obtaining a truly optimal solution presents a challenge.

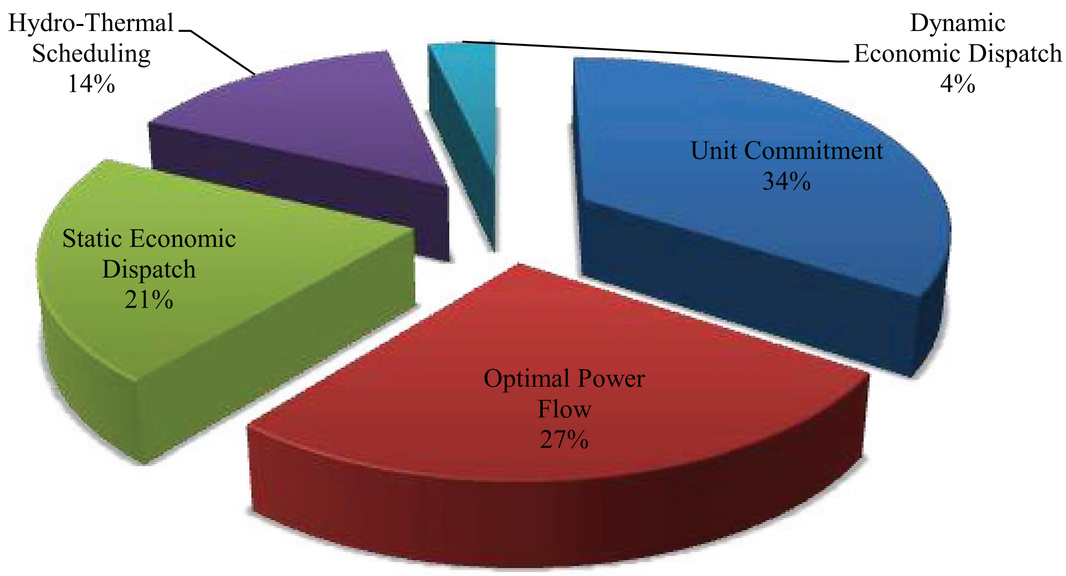


Figure 1. 1: Division of published papers (~1700) for different electric power system optimization areas: operating perspective.

Optimization algorithms based on swarm intelligence, known as meta-heuristic algorithms, gained popularity in solving complex and high dimensional optimization problems years ago. Because the performance of most of the meta-heuristic methods are independent of the initial solutions and are derivative-free, they overcome the main limitations of deterministic optimization methods, e.g., getting trapped in local extrema and divergence situations. In addition, the characteristics of the objective function and/or constraints are inconsequential for the success of those algorithms. Meta-heuristic algorithms are easy to implement and can be combined with others. Therefore, most researchers have been inspired to combine two or more methods to offer an efficient hybrid optimization method. The core reason behind hybridization is to enhance the solution quality by overcoming the limitations of each technique. Although initiation of meta-heuristic algorithms uses random values, they follow a logical pattern. Deep inside most meta-heuristic algorithms, two procedures (exploration and exploitation) interact with each other. Attaining a global solution is not guaranteed by these methods; hence they simulate independent runs to support their achievements. The efficiency of these algorithms is confirmed by statistical measurements. One drawback of meta-heuristic algorithms is the adjustment of their parameters which follows a trial-and-error process and varies from one application to another. An algorithm with few parameters to be tuned is always favored, as long as it is efficient.

Optimization problems can be classified into constrained and unconstrained types. The former includes most of the practical and real-world applications that are commonly solved via independent constraint-handling techniques. The effectiveness varies from one technique to another and with different algorithms. New meta-heuristic optimization algorithms provide another direction for future research, but outperforming the previously introduced methods is not an easy task. Solving several benchmark optimization functions and real-world optimization problems are commonly the practices to emphasize the potential and efficacy of any new algorithm.

1.2 Thesis Objectives

The main objective of this thesis is to attain practical algorithms for optimal economic and environmental operation of electric power systems. Because this objective can be interpreted in a mathematical optimization problem, it is formulated as a multi-objective optimization problem – one in which nonlinear and non-convex characteristics are more prevalent. As conventional optimization methods are experiencing local solutions' attainments or divergent, meta-heuristic optimization methods increase in popularity because they outperform conventional ones in such complicated problems. A state-of-the-art review of the DED problem is also considered.

One goal of this thesis is to study, understand, and implement the artificial bee colony (ABC) algorithm to solve one of the complex real-world engineering problems. The ABC algorithm is a recently-introduced (November 2005) unconventional optimization method. It is a population-based technique inspired by the intelligent foraging behaviour of the honeybee swarm. A comprehensive survey of the literature that utilizes the ABC algorithm is also offered in this thesis.

Inspired by the intelligent behaviour or survival tactics of a sensory-deprived human being, a new meta-heuristic optimization algorithm is proposed. The sensory-deprived optimization algorithm (SDOA) is based on a solid concept utilizing the exploration and exploitation processes simultaneously and distinctly from other meta-heuristic algorithms. Its distinct and main advantage is in the semi-exploitation and semi-exploration processes, covering a wide range of the solution search-space, and avoiding premature convergence. A temporary parallel semi-exploration routine enhances the solution of the main algorithm. Its efficiency is accentuated after solving several benchmark optimization functions as well as the DED problem. The results are compared with those attained utilizing other well-known optimization methods.

The ABC and SDOA are utilized in this thesis to solve the optimal economic and emission dispatch problems. An attempt to integrate a renewable source and analyze its impact is conducted here as well. A new constrained search-tactic that enhances both algorithms' performance is proposed. Various test systems are adopted to verify the effectiveness of these suggested methods. The promising outcomes from both algorithms accentuate their potential to be applied in other electric power system optimization areas.

1.3 THESIS CONTRIBUTIONS

The following list highlights the contributions of this thesis:

1. Offering a state-of-the-art overview of the DED problem. Various techniques are used to classify the reviewed literature into three categories. Advantages, disadvantages, and considered constraints of each paper are highlighted.
2. Studying, analyzing, and implementing the ABC algorithm in solving complex mixed integer nonlinear optimization problems such as the optimal allocation of a distributed generation (DG) application.
3. Presenting a comprehensive survey of the literature that employed the ABC algorithm and categorizing these areas of application.
4. Proposing a novel meta-heuristic optimization algorithm. The SDOA is inspired by the intelligent behaviour or survival of a sensory-deprived human being. A set of benchmark optimization functions is examined to confirm the enhancement of the performance of the suggested algorithm. In addition, the results obtained are compared with those from other well-known meta-heuristic optimization algorithms.

5. Applying the SDOA in solving a mixed integer nonlinear optimization problem, i.e., the optimal allocation of DG application in distribution systems.
6. Utilizing the ABC algorithm in solving the DED and the dynamic economic emission dispatch (DEED) problems, i.e., the single and multi-objective functions respectively, using different test systems.
7. Implementing the proposed SDOA in solving the dynamic economic dispatch problem. In addition, analyzing the potential of the SDOA algorithm in complex, highly nonlinear, and non-convex optimization applications.
8. Integrating a renewable source in solving the DED and DEED problems, and emphasizing the impact of such integration.
9. Proposing a new constraint-handling strategy in solving the DED and DEED problems. The effectiveness of this strategy is measured using different meta-heuristic algorithms and test systems.

1.4 THESIS OUTLINE

This thesis is organized and divided in six chapters as follows. The motivation, objectives, and contributions are highlighted in this first chapter. The second chapter provides an overview of the dynamic economic and emission dispatch problems considered. It also emphasizes and categorizes the optimization tools used in the literature. The third chapter introduces one of the modern meta-heuristic optimization algorithms utilized in this thesis – the ABC algorithm. A summarized survey of the literature employing the ABC algorithm as well as categorization of these application areas are presented in chapter three. The novel meta-heuristic algorithm (SDOA) is described in the fourth chapter and a set of benchmark optimization functions is examined to confirm its efficiency. The fifth chapter demonstrates the mathematical formulations and computational results of the problems considered. In addition, it highlights the impact of integrating a renewable source in both objective functions, and verifies the effectiveness of the new constrained search-tactic. The final chapter states the conclusions and recommendations for future research. List references and an appendix end this thesis.

CHAPTER 2: OPTIMAL OPERATION OF ELECTRIC POWER GENERATION

2.1 INTRODUCTION

The dynamic economic dispatch (DED) occupies a prominent place in a power system's operation and control. The goal of DED is to determine the optimal power outputs of on-line generating units in order to meet the load demand subject to satisfying various operational constraints over finite dispatch periods. In practice, there are static economic dispatch (SED) and DED problems. The latter considers additional practical constraints such as upper and lower bounds on the units' ramping-rates. In reality, units will not respond to steep or instantaneous load variations. Early research works responding to this aspect were published in the 1970s [1, 2].

Optimal operation of electric power system networks is a challenging real-world engineering problem. As shown in Figure 2.1, many optimization problems interact in such a way that the solution obtained from one problem (block) depends on the outcome of an adjacent block's solution. In general, iterative procedures are used to integrate those problems. Survey papers dealing with these optimization problems include [3] which summarizes various problem-solving techniques utilized for the UC problem. Furthermore, the OPF problem has been reviewed in [4-6]. The SED – known alternatively as an economic dispatch (ED) – has been addressed in [7, 8].

The DED problem has not been addressed as thoroughly as other optimization problems. In Feb. 2010, [9] offered a review of optimal dynamic power dispatch highlighting the mathematical formulations and some available solution techniques. The present work is different in that it categorizes earlier work based on the optimization algorithms. Figure 2.2 illustrates annual published research on "Dynamic Economic Dispatch" according to a review of the IEL and Science Direct databases. The hydro-thermal scheduling problem is not considered in this work because [10] offers a comprehensive survey of that topic. This chapter is organized as follows: Section 2.2

summarizes the reviewed literature and offers a categorization of problem-solving techniques into three groups. Section 2.3 is the conclusion.

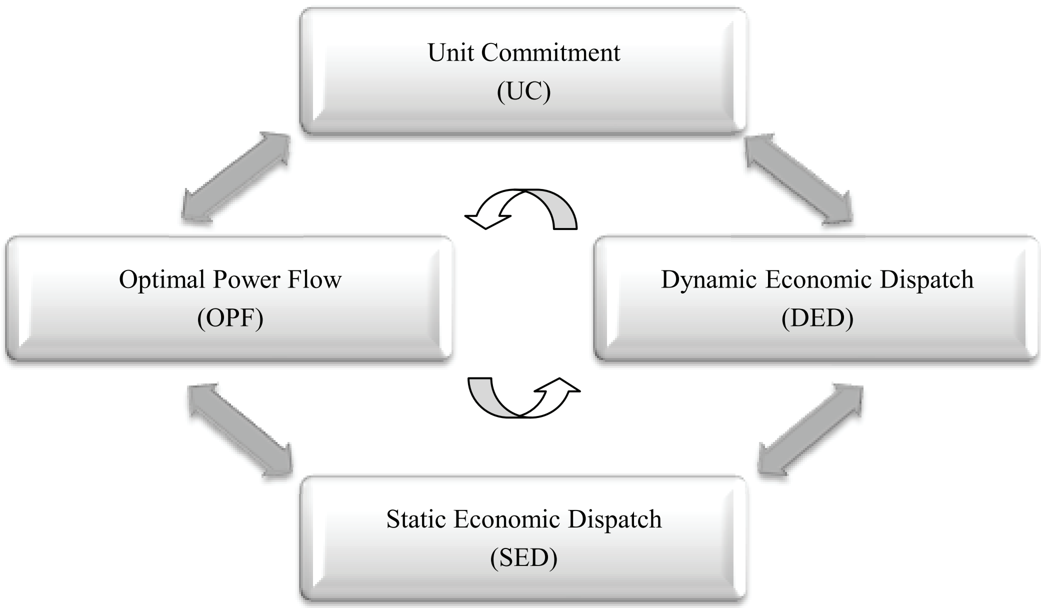


Figure 2. 1: Interactions between operational power system optimization problems.

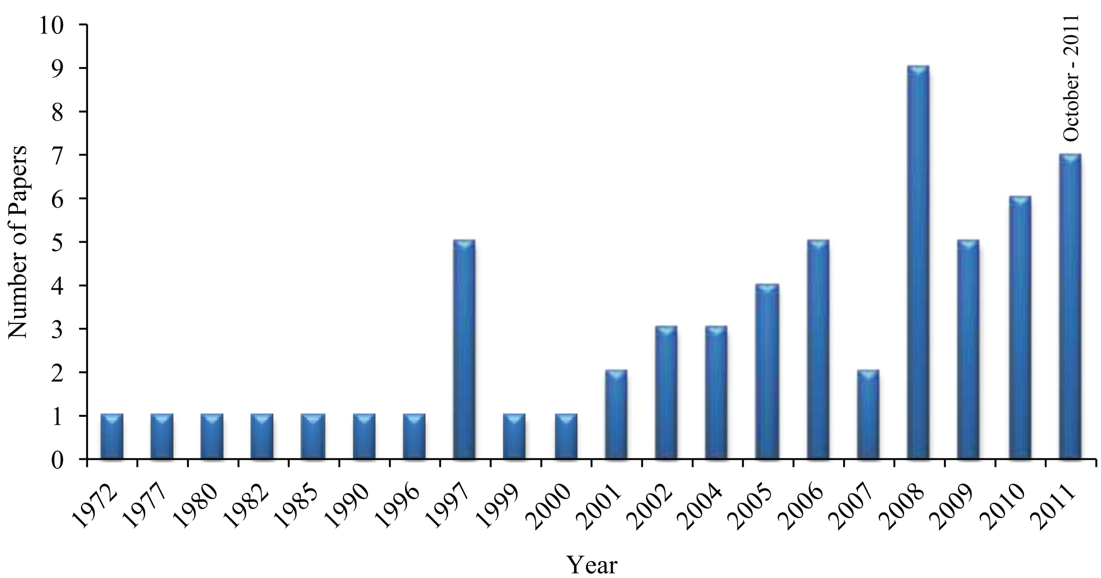


Figure 2. 2: Distribution of publications considering the dynamic economic dispatch problem per year.

2.2 A LITERATURE REVIEW

The DED formulation is considered to be an accurate and practical model for dispatching on-line units economically. However, obtaining a truly optimal solution presented a challenge as some computational methods do not yield a global minimum as many local extrema may exist.

Since the DED problem was first recognized, deterministic algorithms such as variational solutions using Lagrange multipliers, dynamic programming (DP), nonlinear programming (NLP), linear programming (LP), quadratic programming (QP), and sequential quadratic programming (SQP) have been among the proposed techniques. However, heuristic algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), evolutionary programming (EP), differential evolution (DE), Tabu search (TS), and simulated annealing (SA) seem to have shared the same dominance as deterministic algorithms. This is because heuristic algorithms, unlike deterministic ones, are derivative-free, and capable of solving optimization problems without requiring convexity. They are also independent of the initial solution, and have the ability to avoid being trapped in local optima. On the other hand, heuristic algorithms have drawbacks such as being problem dependent, requiring parameter tuning, and unable to guarantee global solution attainment. Therefore, a variety of research efforts were directed at combining more than one technique into a single (hybrid) algorithm such as LP-QP, EP-SQP, GA-SA, and PSO-SQP. The idea behind the hybridization is to enhance the solution quality by overcoming the limitations of each individual technique.

The techniques in this chapter are divided into three categories as deterministic, heuristic, and hybrid methods. Figure 2.3 and Figure 2.4 demonstrate the trend of published research based on different techniques based on a review of the IEL and Science Direct databases. Clearly, hybrid methods account for the majority of currently published algorithms.

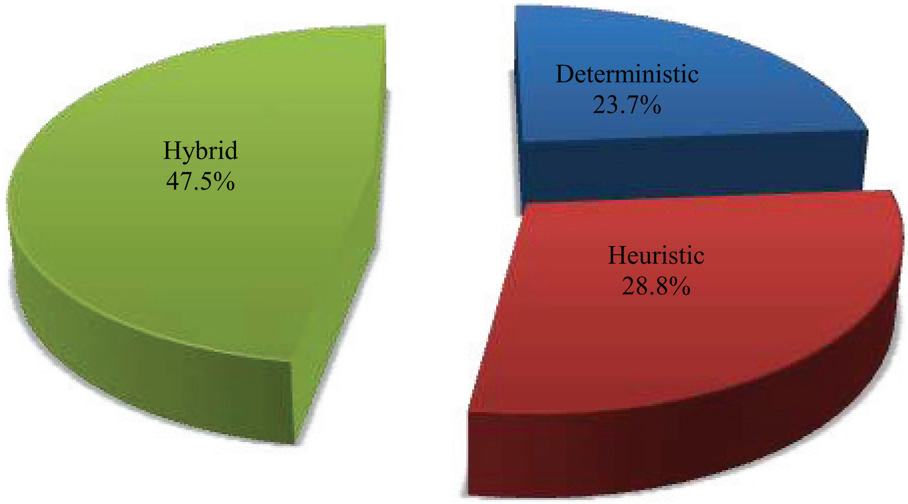


Figure 2. 3: Division of methods of published papers used in solving the dynamic economic dispatch problem.

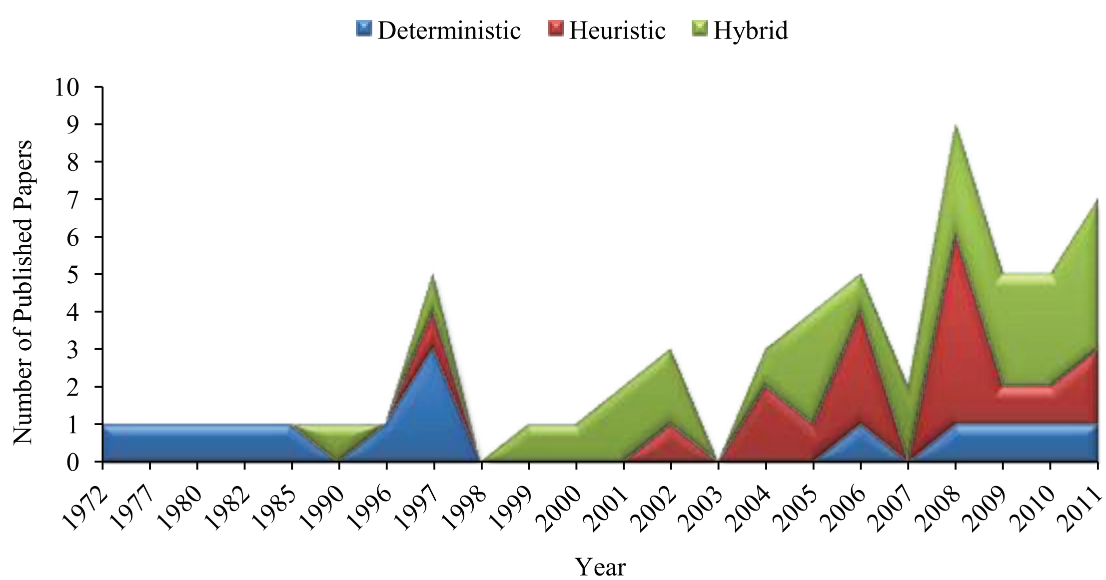


Figure 2. 4: Trends of published papers considering the dynamic economic dispatch problem per year.

2.2.1 Deterministic Methods

Most of the early methods proposed to solve the DED problem used deterministic techniques such as NLP [11], DP [12], and variational techniques based on Lagrange multipliers [13]. The DED problem was generally solved by discretizing the entire dispatch period into short time intervals, over which the load demand was assumed to be constant. Each time interval was then solved as a SED problem [14].

Bechert and Kwatny [1] were first to include ramp-rate limits of on-line units in optimal dynamic dispatch of thermal generation. They proposed an analytical approach and combined the economic load allocation and supplementary control action into a single dynamic optimal control problem. An optimal feedback controller was computed using Pontryagin's maximum principle.

A multi-pass DP technique was offered in [2] while considering the valve-point loading effects for up to five generators. The system was modeled using state equations where the units' output and rate of change of the generator power output were the state and control variables, respectively. Both generators outputs and time-stages were normalized to 32 points. The proposed technique started with a coarse-grid and then switched to a finer-grid to reduce the problem's dimensionality.

The NLP algorithm was utilized in [11] to solve the DED problem taking into account the system's spinning reserve constraints. The author suggested that penalty factors derived from generator ramp-rate constraints be included in the objective function. However, the formulation neglected the transmission losses and valve-point effects, which in turn degraded the practicality of the method.

Ref. [12] developed a procedure based on successive approximation DP to solve the problem by dividing it into stages. Each stage was solved using forward DP considering two units at a time. Three pairing schemes were suggested but no particular scheme was recommended as offering the best convergence results. First, a circular-pairing scheme was offered, where units were indexed based on their incremental cost curve ranked from the "least expensive" to "most expensive". The following sequence describes the second scheme referred to as the spiral-pairing scheme. The "least expensive" unit is paired with the "most expensive" one; then, the "most expensive" unit is paired with the second "least expensive" one; after that, the second "least expensive" unit paired with the second "most expensive" unit; and so forth. A combination of the two earlier schemes represents the third pairing scheme.

The proposed methods in [1, 2, 12] neglected the transmission losses and spinning reserve constraints. On the other hand, Wood in [13] considered these constraints in solving the reserve constrained DED problem while neglecting the units' valve-point effects. The proposed technique divided the DED problem into SED sub-problems, which were then solved in a backward sequence employing the Lagrange multiplier technique. Quasi-optimal solution was the main drawback of the suggested algorithm.

The LP technique was employed in [15] to solve the DED problem with security and emission inequality constraints included via the penalty function method. Five test cases were examined to demonstrate the effect of the considered constraints with a dispatch period of 12 intervals. A quadratic fuel cost function was used while neglecting valve-point effects and spinning reserve requirements.

Based on DP, a quasi-static economic dispatch method was presented in [16]. In addition to solving the SED problem, the proposed method provided quasi-optimal initial starting points for the DED problem. The adopted fuel cost function (objective function) was not typical. Furthermore, the objective function was subject to the unit's ramp-rate and output power limits only. The authors claimed additional constraints could be added to the objective function by means of penalty factors.

The authors of [17] proposed an algorithm for solving the multi-stage DED problem. The algorithm was an extended version of their previous work of [18]. The suggested algorithm included ramp-rate constraints and multi-stage periods, i.e., two-stages. The adjacent (multiple) stages were included in the objective function via Lagrange multipliers. The system's security constraints were considered in the formulation, but the generator units' valve-point effects were neglected.

The authors of [19] analyzed the effect of dynamic constraints on optimal short-term transactions in a deregulated environment utilizing successive DP methods. The security of the system was considered through a DC power flow model, but the effect of the transmission losses was neglected. The time of exporting and/or importing power was included as inequality constraints. The dynamic constraints have a significant impact on the scheduled transaction and operational cost. In other words, considering such short-term transactions constraints may change the amount and reverse the direction of the transactions for optimality.

Chandram *et al.* [20] utilized the Muller algorithm in order to solve the DED problem. The proposed algorithm involved the selection of lambda values. Then, the generators output powers were expressed in terms of the lambda by interpolation, and the actual value of lambda calculated from the power balance constraints using Muller's method. Two test systems were examined, and the results of the proposed method were reported to outperform those obtained using competing methods with respect to the central processing unit (CPU) time requirements. The inequality constraints contained the unit's ramp-rate and output power limits only. The transmission losses of the system were neglected, and the fuel cost function was convex.

The Brent method – a root finding method [21] which involves root bracketing, inverse quadratic interpolation, and bisection techniques – was used in [22] to solve the DED problem to determine the incremental fuel costs for all units committed without taking into account valve-point effects. In other words, the minimum and maximum lambda values were calculated based on the unit's minimum and maximum power output. The optimal lambda at a predetermined load value was obtained using the Brent method. Although the transmission losses of the system were considered in the solution algorithm, the spinning reserve requirements were neglected. Two systems were adopted to evaluate the proposed method's efficiency, and the results were compared with the lambda iterations technique. Although the suggested method required less CPU time, the operational costs obtained were only slightly less than those obtained using the lambda iteration technique.

The authors in [23, 24] suggested that the differences between the DED and optimal control dynamic dispatch (OCDD) models were that the optimal solution of the former was based on optimization theory, independent of the value of the initial solution. On the other hand, the optimal solution of the latter depended on the initial solution value, and was based on control theory. The authors extended the DED formulation by introducing additional constraints so the anticipated repeated implementation did not violate the units' ramp-rate constraints. Accordingly, a model predictive control (MPC) based on OCDD was offered. As additional constraints were added, the operating cost of the proposed algorithm was higher than that obtained using the DED formulation. The transmission losses of the system and spinning reserve were neglected. In addition, the quadratic fuel cost function was used without considering the units' valve-point effects.

Hemamalini *et al.* [25] applied the Maclaurin series-based Lagrangian method to solve the DED problem. The proposed method utilized the representation of a valve-point effect, i.e., the rectify sinusoid term in the fuel cost function, for an approximated Maclaurin series expansion. An additional factor term (tuned via trial-and-error process) has been embedded to overcome the approximation error. Three test systems with valve-point effect, spinning reserve, and transmission losses were used to evaluate the proposed method. Significant reduction in the required CPU time was the main advantage of this method, although some of the results using other methods achieved lower minimum fuel costs.

2.2.2 HEURISTIC METHODS

The DED problem was one of the real-world optimization problems that has benefited from the development of the heuristic algorithms. Most of these algorithms were population-based, relying on initial randomization associated with logical patterns. Different constraint handling methods were suggested for heuristic algorithms [26]. As an alternative to the well-known penalty function constraint handling method, the author of [27] suggested a feasibility-based selection comparison technique to handle the problem's constraints. It employs a tournament selection operator, where two solutions are compared at a time when the following conditions are imposed: 1) any feasible solution is preferred over an infeasible one, 2) among two feasible solutions, the one with better objective function value is preferred, and 3) among two infeasible solutions, the one having the smaller constraint violation is preferred. This technique, reported in [28] and [29], offered promising results.

Ongsakul and Tippayachai [28] suggested a parallel micro GA based on merit order loading solutions to solve the DED problem. The proposed algorithm was implemented on an eight-processor scalable multicomputer. The initial population was divided into eight species performing in parallel. Species numbers two to seven were slaves, and species number one was set to be the host, i.e., used for sending and receiving data from slave processors. The transmission losses were considered in the power balance equation, but the units' valve-point effects and system's spinning reserves were neglected.

The PSO algorithm was utilized to solve the DED problem in [30]. The units' valve-point effects were represented by prohibited operating zones. The system spinning reserves, transmission losses, and security constraints were included in this work as well. The proposed algorithm examined two test systems over a 24-hour dispatch period. However, the dispatch period was divided into intervals of one hour each which were solved sequentially. The results outperformed those obtained using other methods (namely fast and improved fast EP techniques) with respect to required CPU time and solution quality.

In [31], a multi-objective optimization problem was formulated. The two competing functions were the fuel costs and emissions, known as the dynamic economic emission dispatch (DEED) problem. In addition, the units' valve-point effects and transmission losses were considered in the proposed approach. Moreover, the author assumed that the decision-maker has a goal (target) for each objective function. These two functions were transferred to a single objective function using a goal-attainment method [32]. Finally, the PSO algorithm was implemented to solve the combined objective function.

Basu implemented the non-dominated sorting genetic algorithm-II (NSGA-II) in [33] to minimize the total operating fuel costs and emissions simultaneously. In addition, the units' valve-point effects and transmission losses were considered in the DEED problem. The author offered a set of Pareto-optimal solutions to the decision-maker in a single run, as a substitute to retaining one optimal solution using the linear combination method. Therefore, an advantage was attained as the proposed method required less CPU time with respect to the compared one.

A bid-based DED utilizing a PSO algorithm to maximize the social benefit in a competitive electricity market was proposed in [34]. The authors considered bids from both supply and demand sides. In addition, fuel emissions, security, and transmission losses were included in the problem's constraints. Two bid-based trading periods were considered and the objective function solved, initially unconstrained. Then, constraints were added one at a time to observe the effect of each. Clearly, as many constraints were included in the optimization problem, the social benefit was decreased.

An elitist GA approach was proposed to solve the DED problem [35]. The best solution in each generation was retained to pass the benefits to subsequent generations. The cost function of each unit – based on the adopted system – was represented by a linear function, and the transmission losses were included in the equality constraints. Moreover, the dispatch time period was discretized into 48 intervals, each of which was solved successively. The authors analyzed the ramp-rate effect on the operating economy based on the assumption that the ramp rates were taken as percentages (from 10% to 50%) of the previous power output. The authors stated that the stricter the ramp-rates, the higher the operating cost.

A constrained EP technique was utilized to solve the DED problem in [36]. Valve-point effects were included in the generators' fuel cost functions, but the transmission losses were neglected. The proposed technique forced the initial population to be feasible. In other words, the constraint's violations were corrected using an iterative process. In order to generate a new population, the following procedure was suggested. At a randomly selected time, two generators were chosen arbitrarily, and then a fraction of the output power from one to the other was transferred while the constraints were not violated. Such an approach would require large CPU time, especially for large systems.

The DED problem has been solved in [37] utilizing the EP technique. A designated system has been tested for static and DED problems. In both tested problems, variations of two loads buses were solved separately. Although the authors recorded the transmission losses, they did not include it in the equality constraints, i.e., power balance equation. Thus, constraints were handled, seemingly via repeated power flow calculations. Implementation of the DED resulted in higher operating fuel costs compared to SED.

The authors in [38] solved the DED problem using the SA technique. The units' valve-point effects and transmission losses were taken into account in the solution. The proposed method was examined on one test system, and required approximately six minutes to attain a solution. The authors suggested that using such an algorithm on parallel processing platforms would reduce the required CPU time.

The enhanced real-parameter quantum evolutionary algorithm (QEA) was proposed to solve the DED problem accounting for valve-point effects in [39]. Moreover, two heuristic techniques have been suggested in this paper to restore a feasible region and to improve the solution quality. In other words, the feasibility restoration attempt heuristic (FRAH) and the incremental cost-based optimization heuristic (ICH) techniques were invoked in the solution process. The first technique constrained the population to start at the interval with the largest demand and moved to the next extreme ones taking into account the problem's constraints. Once the feasible region (solution) was restored, the second process was initiated in order to attain an enhanced solution by local search. Two test systems were considered, with and without the units' valve-point effects. However, the transmission losses were neglected in both systems. In particular, the results of the proposed method when units' valve-point effects were considered outperformed those obtained using alternative methods.

A multiple Tabu search (MTS) algorithm was employed to solve the DED problem in [40]. The optimization problem included the transmission losses of the systems, and the units' valve-point effects were represented as prohibited operating zones. To improve the solution of the standard TS algorithm, the authors suggested additional procedural mechanisms, i.e., initialization (to ensure population satisfied the problem's constraints); adaptive search (step size adaptively modified to generating neighbour solutions); multiple searches (via parallel processors); crossover (to enhance the parallel solutions using GA's routine); and re-initialization process (to avoid premature convergence when the search stalled at a local solution). The effectiveness of the proposed algorithm was demonstrated based on statistical measures. The standard deviation and the required CPU time of the proposed algorithm were significantly smaller than those obtained using other methods.

Shang and Sun in [41] proposed a preference-based non-dominated sorting genetic algorithm (PNSGA) to solve the DEED problem. A sample system with simplified assumptions was adopted to exemplify the efficiency of the proposed algorithm. In other words, only one unit was restricted to ramp-rate constraints and the units' valve-point effects were not addressed. Furthermore, the transmission losses of the system were neglected.

A modified version of the DE technique (MDE) aiming to solve the DED problem was proposed in [42]. The authors adopted a feasibility-based selection comparison technique to handle the problem's constraints which were derived from three selection criteria and inspired by [27]. The units' valve-point effects were considered in the fuel cost function. However, the transmission losses of the system were neglected. The proposed MDE method obtained additional cost savings as compared to other methods. The required CPU time to attain a solution was relatively shorter than those methods.

A variable scaling hybrid differential evolution (VSHDE) algorithm was proposed in [43] to solve the DED problem. This method considered the units' valve-point effects, the transmission losses of the system, and spinning reserves. In addition, a prohibited operating zone was included in the problem constraints. The penalty function method was used to convert the constrained DED problem to an unconstrained one. However, the prohibited operating zone was handled using a delimitation point that divided the zone into two sub-zones. Consequently, the unit was forced to adjust its output to the rated limit once it operated in a prohibited sub-zone. This was done based on a comparison between a uniformly distributed number [0,1] and the value of the delimitation point. To overcome the limitation of the hybrid DE (HDE) algorithm, the author integrated two operators (migrating and accelerating) in the solution procedure. The migrating operator maintained the diversity of the individuals, and the accelerating operator was embedded to accelerate the algorithm performance. In addition, the scaling factor was updated in each cycle, based on the probability of the population, using a one-fifth success rule. Two test systems were considered for a one-day dispatch period, and the required spinning reserves were set larger than 5% of the load demand at every dispatch hour. The results were compared with those obtained using the HDE algorithm and a significant reduction in operating costs was illustrated. Although the VSHDE reduced (relatively) the required CPU time to approximately 50%, it consumed 7 and 23 hours for 10 and 20 unit systems respectively.

The pattern search (PS) algorithm was used in [44] to solve the DEED problem. The fuel cost function included a rectified sinusoid term that represented the units' valve-point effects. The units' ramp-rate limits as well as the transmission losses were included in the inequality and equality constraints respectively. However, the paper neglected the spinning reserves constraints. The main goal was to solve the DEED problem for a multiple dispatch periods, i.e., two days scheduling without violating the ramp-rate constraints in transition between the days periods. The adopted system was used for both single and multi-objectives optimization problems. The two conflicting objective functions were combined as a single one with equal weights. In addition, the results of the proposed method were compared with those obtained using the SA and EP techniques.

The artificial immune system (AIS) was adopted in [45] and [46] to solve the DED problem. The AIS algorithm uses four iterative strategies, i.e., generating (random) population known as the immune cells or antibodies; cloning each member of population based on the affinity rate identified as the proliferation; mutating mechanism recognized as the maturation process; and eliminating antibodies showing no improvement or are trapped in local solutions called the aging operator. The performance of the proposed method in [45] was evaluated by solving a single test system, taking into account the units' valve-point effects and neglecting the transmission losses of the system. On the other hand, the authors of [46] tested two systems considering the units' valve-point effects. One of the two systems was solved by neglecting the transmission losses. The outcomes from the AIS method in both references were better than those attained using other algorithms in terms of solution quality and computation time.

2.2.3 HYBRID METHODS

Although most of the heuristic and deterministic algorithms successfully obtained some solutions to the DED problem, they have drawbacks – as stated earlier in this section. Therefore, a combination of more than one technique has been proposed in the literature to solve this. Hybridization commonly divided the process (optimization) into two phases. The first phase, designated for the heuristic algorithm, aimed to explore the search-space without restrictions. In the second phase with a potential region discovered in the primary phase, a deterministic algorithm took over seeking to enhance the results. This routine is repeated until a termination criterion was met.

A combination of GA and a gradient search method was proposed in [14]. The GA was assigned the initial direction of the search towards the optimal region. Subsequently, the gradient method was applied for local search in that region. The authors utilized a smooth quadratic fuel cost function. The transmission losses of the system were considered, however the units' valve-point effects were neglected. Additional constraints were added in the local search cycle. In other words, the amount of perturbation for any randomly selected unit was identically applied to the remaining units – but, in the opposite direction (negative sign). The authors divided the dispatch time period (one day) into 48 intervals, and in each interval the DED problem was solved as a SED problem. In addition, two hybrid GAs were suggested, i.e., with predetermined and flexible local search time, respectively. Although the solution quality of the second hybrid algorithm outperformed the first one, the execution time was relatively long and the results obtained by both approaches outperformed those obtained via the conventional GA technique.

An improved differential evolutionary (IDE) combined with the Shor's r-algorithm was proposed to solve the DED problem in [29]. Once more, the optimization process was divided into two parts. The first part or base part was assigned to the IDE, and the fine tuning of the retained solution from the IDE was the task of the second part. The idea of Shor's r-algorithm is to make steps in the direction opposite to a sub-gradient at the current point, i.e., by calculating the difference between the current and previous sub-gradient points per iteration [47]. Although the objective function included the units' valve-point effects, the transmission losses of the system and spinning reserves were neglected. As an alternative to the classical constraints handling method (penalty function), the authors adopted the feasibility-based selection comparison method. The single test system was examined, and the results of the proposed algorithm outperformed those attained using competing methods. Despite that, unit-2 output power at the 20th dispatch hour violated the unit's bounded constraints.

The authors of [48], [49], and [50] proposed a solution algorithm for solving the DED problem considering valve-point effects as a rectified sinusoid component integrated into the cost function. In all suggested hybrid algorithms, the authors divided the solution process into two parts. The first part applied a heuristic algorithm (PSO, EP, and deterministically guided PSO respectively) where constraints were relaxed. The second part exploited a deterministic technique (SQP) to enhance the solution obtained by the heuristic methods taking into account the problem's constraints. In other words, the heuristic algorithm explored the search space with relaxed constraints and once the cost function of the current iteration was better than that of the previous one, the SQP routine started. The improved results of SQP was then retained as a new starting point for the next heuristic(s) search. This sequence was repeated for a predetermined number of iterations. In addition, the authors examined three test cases with different load curve patterns. In modified hybrid EP-SQP [49] and deterministically guided PSO [50] the transmission losses have been evaluated using the *B*-coefficients loss matrix, while in PSO-SQP [48] the power flow calculation has been utilized to calculate the transmission losses of the system, spinning reserves, and security constraints.

The author of [51] proposed a hybrid EP and fuzzy satisfying approach to solve the DEED problem. Its mathematical formulation was identical to the one in [31]. However, in [51] the two competing objective functions were modeled applying fuzzy sets, with the assumption that the decision-maker has a fuzzy-goal per objectives. The decision-maker defined the reference membership values and the problem was solved using an EP method. Moreover, the results of the proposed method were compared with those obtained using a fuzzy satisfying method based on the SA technique.

An extended version of the interior point quadratic programming (IP-QP) technique was proposed in [52] to solve the bid-based DED problem. The solution was obtained assuming that the bid price curves for generators and consumers were quadratic-convex and quadratic-concave functions, respectively. Moreover, the emission effect was considered in this reference as an inequality constraint. The transmission losses of the system were included in the equality constraints; however the issue of spinning reserves was not addressed. The objective function dealt with a competitive electricity market to achieve maximum profits.

A hybrid QP-LP process was suggested in [53] to solve the DED problem. Three problems were defined, and the constraints for each coupled through the units' ramp-rate limits. The proposed method initially solved the base case SED by relaxing the ramp-rate constraints using a QP algorithm. Then, the LP technique was utilized to solve the dispatch problem considering the ramp-rate constraints. The latter technique linearized the optimization problem about the former base case results. In other words, the objective function and constraints were linearized. The transmission losses of the system, security, and spinning reserve were included in the problem constraints. However, the units' valve-point effects were not addressed.

Based on artificial neural networks (ANN), the author of [54] proposed an algorithm with two stages to solve the DED problem. First, the lambda-iteration technique employed to yield the SED (base case) result by relaxing the ramp-rate constraints. Second, the Hopfield neural networks (HNN) employed to solve the DED problem. The neural networks (NN) included in a closed-loop structure so that, when a constraint violation occurred, the magnitude and direction of the violation were fed back to adjust the neurons' states. The optimization methodology was adapted from [53], but the security and transmission losses constraints were neglected in this paper.

Li and Aggarwal in [55] combined a relaxed GA and gradient technique in order to solve the static and dynamic ED problems. The proposed method modified the hybrid GA, which was offered by one of the authors in [14] to enhance the solution quality and accelerate the algorithm performance. In other words, the power balance constraint was relaxed in the base search, and adjusted at the beginning of the gradient technique (local search). The authors reported that the wider feasible region offered by the relaxed approach obtained a potential solution with a likelihood of shorter time than a stricter approach. Although the transmission losses of the system were considered, the units' valve-point effects were not measured.

Two solution methods for the DED problem were proposed in [56]. In both methods, the operating cost of each interval was optimized separately. To anticipate the difficulties arising from such separation, the authors considered intervals ahead of the current interval by defining a set of unit parameters and system variables. The aim of the first method was to find a feasible (quasi-optimal) solution suitable for different load profiles via an adaptive look-ahead technique. The second method, using deterministic (QP or LP) algorithms, attempted to find the optimal solution. In addition, the QP and LP methods were suggested to solve the optimization problem for quadratic and linear cost functions respectively. For simplicity, the authors neglected the transmission losses of the system and spinning reserves. The units' valve-point effects were omitted.

In Ref. [57] a combination of the SA technique and GA to solve the DED problem considering the transmission losses and different types of units' cost functions was suggested. However, the units' valve-point effects and system's spinning reserves were neglected. Each dispatch period was optimized individually via SED. Utilizing the SA provided a sufficient initial solution for the GA cycle. Increasing and daily load demands were considered. However, the average CPU time (per time period) of the proposed method was relatively high by comparison.

The authors of [58] proposed a hybrid method divided into two parts to solve the DED problem. The first part utilized the EP technique to obtain a quasi-optimal region. Subsequently, the second part was developed for local search to determine the optimal solution by applying the SQP technique. Two case studies were considered. The first one was to observe the impact of population size on the solution quality using EP technique only. The second employed the proposed hybrid EP-SQP method and the results were compared using EP and SQP individually. In both cases, the units' valve-point effects were integrated with the fuel cost function. However, the transmission losses of the system and spinning reserve requirements were neglected.

The authors of [59] proposed a fuzzy optimization technique to solve the DED problem under the conditions of an uncertain deregulated power environment for a 10-minute reserve market. Moreover, a goal satisfaction concept based on a decision-maker's option was used in the optimization process. The uncertain parameters were represented by fuzzy numbers, and comprised of reserve required, prices cleared, and the probability that reserves were called in actual operation. The objective function was to maximize the generation company (GENCO) profit. However, the units' valve-point effects and transmission losses were neglected.

A real coded GA combined with the quasi-simplex algorithm has been proposed in [60]. The genes of the proposed method were represented using decimal digits. In addition, three different rules (proportion, minimum cost, and LP) have been offered in order to provide a sufficient initial population. One test system has been considered and the units' valve-point effects were absorbed in the cost function. However, the transmission losses of the system and spinning reserves were neglected. Moreover, the authors did not consider solutions obtained from other methods, and the efficiency of their method relied on the fact that the standard deviation of the proposed method (after conducting 10 independent runs) was relatively small.

Ref. [61] proposed a solution to the thermal-wind coordination scheduling problem. The procedure was divided into stages, but the transmission losses of the system and units' valve-point effects were neglected. The first one utilized the SA technique to schedule the generating units, but when considering the wind units the constrained DED was solved by the second stage, i.e., using the direct search method. The author divided the dispatch period into intervals, each of which was solved successively. The operating costs considered in this reference included two terms: 1) the second-order polynomial function representing the fuel cost and, 2) the start-up cost of a unit. Due to the intermittency of wind power generation, the system's spinning reserve was calculated as a simple fraction of the predicted wind power generation. The load supplied by the thermal units was the net-load, which was solved by the direct search method. In other words, the load supplied by the thermal units at a specified time was reduced by the amount of wind power generation at that time. There was significant reduction in the operating costs when optimal wind power generation was integrated in the system. Although wind power generation has no cost that varies with the power output, it would require an additional system spinning reserve (due to the intermittency of the wind resource) which would increase the operational cost. The trade-off of emissions' constraints may guide the decision-maker in such a situation.

A combination of HNN and QP was proposed to solve the DED problem in [62] and [63]. The problem was divided into two phases; the first phase was assigned to the HNN to solve the SED part, and the second phase used the QP method to solve the DED part. Furthermore, a set of procedures (backward/forward) was suggested and then applied to consider the ramp-rate constraints. A correction factor – based on the power balance mismatch – has been offered to accelerate the algorithm in each time interval. Although the authors included the transmission losses in the power balance equality constraints, it was neglected in the tested system as was the system's spinning reserve requirements. The results of the suggested method for a half-day interval were compared with those obtained using other methods. The proposed method attained additional cost saving, and the required CPU time was relatively reduced.

Integrating the swarm direction technique with the fast evolutionary programming (FEP) method was proposed in [64] to solve the DED problem. In addition, the prohibited operating zones, spinning reserves, transmission losses, and line flow limits were considered. The swarm direction technique was applied while creating an offspring, particularly at the mutation stage. A sample system was solved using the proposed algorithm with one-hour and one-day dispatch periods. The results of the algorithm were reported to outperform other FEB-based methods. However, the algorithm required further parameters to be fine-tuned, which in turn added an additional burden to the process.

The method proposed in [65] combined the DE and local random search (LRS) method to solve the DED problem while considering the units' valve-point effects and the transmission losses of the system. However, the system's spinning reserves were not addressed. As a substitute to a constant (user-defined) crossover value, the authors suggested an iteration-variant (adaptive) crossover value. This strategy enabled the algorithm to cover a wide range of the search-space initially, and then narrowed to local search-space to increase the possibility of arriving at an optimal region. Once the adaptive DE reached that region, the LRS process started to fine-tune the solution obtained earlier. The authors adopted the LRS method presented in [66]. It was described as iterative procedures attempting to enhance the retained solution using multiple (neighbours) solutions inside the corresponding search region. Then, a greedy selection mechanism was used to evaluate the new candidate solutions. Additional neighbours (new candidate solutions) were then generated within a narrower space of the corresponding (new) search region. These steps were repeated until reaching the maximum number of iteration. The power balance and units ramp-rate constraints were handled by iterative heuristic procedures. Despite that, the authors adopted the "Death-Penalty concept," introduced in [67], by assigning a large (penalty) multiplier to reject the infeasible solution with respect to the power balance constraints. A sample system was adopted to assess the efficiency of the offered method, and the results were compared with those obtained via various methods. The proposed method achieved enhanced results in terms of total operating costs and less CPU time needed.

A combination of EP, PSO, and SQP was presented in [68] to solve the DED problem. Initially, both EP and PSO methods were utilized to explore the search-space freely. Then after, the SQP was assigned to fine tune the retained solution. Although the units' valve-point effects and prohibited operating zones were considered, the transmission losses of the system were neglected. The results outperformed others such as EP-SQP and PSO-SQP. The authors reported that the additional heuristic algorithm will enhance the solution, but it will add more computational time due to the increase in the number of parameters to be tuned.

The seeker optimization algorithm (SOA) and SQP were hybrid (SOA-SQP) to solve the DED problem in [69]. The result obtained by the SOA was used as an input for the SQP technique. The authors utilized the SOA introduced in [70] as a global search mechanism with the utilization of fuzzy reasoning in determining the length step of the search-space. The SOA names the randomly generated population as the swarm and recognizes their members as seekers. The main strategy of the SOA is to modify the entire population into multiple (three) subpopulations with equal sizes, and seekers share neighbourhoods' information. The designated length step and search directions were utilized to modify the (solutions) seekers. The authors examined the performance of the proposed hybrid method with two test systems taking into account the units' valve-point effects. The transmission losses were considered in one of these systems. However, the required CPU time was not reported in this paper. In addition, the results obtained by the proposed method violated various units' ramp-rate constraints in both systems.

The authors of [71] proposed a hybrid technique to solve the DED problem by combining the bacterial foraging method (BF) as a local search and PSO-DE algorithm as the main (global) optimizing mechanism. The BF algorithm introduced in [72] mimics the survival of bacteria in a changing environment. The BF employs iterative steps of fitness evaluation based on food searching and motile behaviour. A sample system, considering the units' valve-point effects and neglecting the transmission losses of the system, was adopted to evaluate the performance. The BF-PSO-DE method outperformed those attained using various methods. The proposed method has been adopted by the same authors in [73] to solve the DED problem considering the transmission losses, security, and spinning reserve constraints. The power flow calculation was used to maintain the equality constraints. One designated system was used to evaluate the performance of the suggested hybrid method, and a comparison with results obtained using different methods was conducted.

A combination of the PSO and harmony search (HS) algorithms was used in [74] to solve the DED problem. The HS method introduced in [75] imitates the musicians' improvisation process via an iterative routine. These procedures can be summarized as: generating the initial harmony memory (population); improvising, i.e., modifying the harmony memory using a set of rules; evaluating the harmony memory based on fitness value; terminating the process if the maximum number of iteration is reached. As an alternative to the classical HS algorithm, the concept of an improved HS (IHS) version suggested in [76] was adopted. The HIS, in this paper, used an adaptive pitch adjustment rate (PAR) to modify the harmony memory based on a PSO mechanism. The authors in [74] examined the proposed method with three test systems considering the units' valve-point effects. However, the transmission losses were only taken into account in the first test case. The average CPU time required to solve those systems were 2.800, 12.233, and 27.650 minutes respectively. In addition, the results of the PSO-HS outperformed those attained using other methods.

The authors of [77] proposed three versions of hybrid chaotic differential evolution (CDE) methods to solve the DED problem. The Chaos sequence exists in nonlinear and dynamic systems, and was defined as a semi-random process created by a deterministic operation via functions known as chaotic maps [78]. In this paper, the Tent's function (chaotic map) [78] was utilized in the three hybrid methods to avoid the parameters' setting and premature convergence of the DE algorithm. The first (CDE1) version utilized chaotic sequences of the Tent's equation to update (adaptively) the mutation and crossover parameters of the DE method. The second (CDE2) version combined the DE, as a main optimizing tool, and a chaotic local search based on Tent's function to avoid premature convergence. In other words, the chaotic local search was initiated prior to the next iteration of the DE method. Therefore, the retained best solution was modified using the designated chaotic map to seek an enhanced outcome. A hybrid of CDE1 and CDE2 represented the third (CDE3) version. The constraints were handled by a set of heuristic rules and procedures. First, the equality (power balance) constraints were adjusted by adding the average mismatch power to all committed units. Then, the units' output power and ramp-rate constraints were tuned at each time interval. The feasibility-based selection mechanism was adopted in all CDE's versions. The performance of the proposed algorithms was illustrated by solving two test cases with both considering the units' valve-point effects but not the transmission losses of the system. Although the results from all three versions were outperforming those obtained using other methods, the CDE3 presented the minimum operating cost among them. The required CPU time of the CDE3 method was slightly higher than those of CDE1 and CDE2 methods.

2.3 SUMMARY

The DED problem illustrated the practical meaning of optimal operation and control of committed generation units to meet the demand of power system networks. However, it considered a complex, non-convex, and nonlinear optimization problem. Some applications raised the complexity of the DED problem by integrating additional (competing) objective functions and/or constraints.

This chapter presented state-of-the-art research of the dynamic economic and emission dispatch problems. Various techniques were used and the reviewed papers were classified into three categories: deterministic, heuristic, and hybrid methods. Advantages, disadvantages, and constraints considered by each paper were also highlighted. Among the techniques, hybrid methods were the most popular due to the significant ease of implementation in solving real-world engineering problems. In addition, combined algorithms boosted the strength and diminished or bypassed the limitation of individual algorithms.

CHAPTER 3: ARTIFICIAL BEE COLONY ALGORITHM

3.1 INTRODUCTION

Optimization algorithms based on swarm intelligence gained popularity in solving complex and high dimensional problems years ago. Because most of the meta-heuristic methods are independent of the initial solutions and derivative-free, they overcome the main limitations of deterministic or conventional optimization methods, i.e., getting trapped in local extrema and divergence situations. The inspiration of most meta-heuristic techniques is natural phenomena, e.g., GA [79], ant colony optimization (ACO) [80], PSO [81], ABC [82], and firefly algorithm (FA) [83].

In addition, meta-heuristic methods are easy to implement and can be combined with other algorithms. The characteristics of the objective function and/or constraints are inconsequential to the success of those methods. However, meta-heuristic algorithms have parameters to be tuned, an adjustment that is commonly accomplished by trial-and-error experiments as well as the skill of the user. Consequently, an efficient algorithm with fewer parameters to be adjusted is always more favorable.

Enhancing performance by accelerating and/or improving the solution quality is an ongoing task for most meta-heuristic algorithms. Various modified versions seeking enhanced results have been suggested in the literature. However, many of those adaptive methods may not succeed depending upon the particular problems and/or additional parameters needing to be tuned.

Optimization problems can be classified into constrained and unconstrained types. The former includes most of the practical and real-world applications; accordingly the constraints of an optimization problem are commonly solved via independent (constraint-handling) techniques [26, 27]. The effectiveness varies from one technique to another and, also, with different meta-heuristic algorithms. Moreover, the superiority of current and future meta-heuristic algorithms can be determined through solving benchmark optimization functions where statistical records emphasize their effectiveness.

The aim of this chapter is to provide a concise survey of one of the newest meta-heuristic methods. The ABC algorithm is a population-based technique introduced late in 2005 [82]. It is inspired by the intelligent foraging behaviour of the honeybees swarm. Vital features and the literature utilizing the ABC algorithm are highlighted. The chapter is organized as follows: Section 3.2 introduces the ABC algorithm and highlights the algorithm's features; Section 3.3 summarizes the literature that employed the ABC algorithm; Section 3.4 outlines the conclusions.

3.2 The Artificial Bee Colony Algorithm

Inspired by the intelligent foraging behaviour of honeybee swarms, the ABC algorithm was introduced to handle unconstrained benchmark optimization functions [82, 84], similar to other well-known meta-heuristic algorithms. An extended version of the ABC algorithm was then offered to handle constrained optimization problems [85].

The colony of artificial bees consists of three groups: employed, onlookers, and scout bees. The employed bees (E_b) randomly search for food-source positions (solutions). By dancing they share information (communicate) about that food source, such as nectar amounts (solutions qualities), with the onlooker bees (O_b) waiting in the dance area at the hive. The duration of a dance is proportional to the nectar's content (fitness value) of the food source being exploited by the employed bee. Onlooker bees watch various dances before choosing a food-source position, according to the probability proportional to the quality of that food source. Consequently, a good food-source position attracts more bees than a bad one. Onlookers and scout bees, once they discover a new food-source position, may change their status to become employed bees. When the food-source position has been visited (tested) fully, the employed bee associated with it abandons it and may once more become a scout or onlooker bee. In a robust search process, exploration and exploitation processes must be carried out simultaneously [82, 86]. In the ABC algorithm, onlookers and employed bees perform the exploitation process in the search-space, while scouts control the exploration process.

One half of the colony size (*CS*) of the ABC algorithm represents the number of employed bees, and the second half is the number of onlooker bees. For every food-source position, only one employed bee is assigned. In other words, the number of food-source positions surrounding the hive is equal to the number of employed bees. The scout initiates its search cycle once the employed bee has exhausted its food-source position. The number of trials for the food source to be called “exhausted” is controlled by the *limit* value. Each cycle of the ABC algorithm comprises three steps: first, sending the employed bee to the possible food-source positions and measuring their foods’ nectar amounts; second, onlookers selecting a food source after sharing the information from the employed bees in the previous step; third, determining the scout bees and then sending them into entirely new food-source positions.

The ABC algorithm creates a randomly distributed initial population of i solutions (i = 1, 2, …, E_b), where i signifies the size of population and E_b is the number of employed bees. Each solution x_i is a D-dimensional vector, where D is the number of parameters to be optimized. The position of a food source, in the ABC algorithm, represents a possible solution to the optimization problem. The nectar amount of a food source corresponds to the quality of the associated solution. After initialization, the population of the positions is subjected to repeated cycles of the search processes for the employed, onlooker, and scout bees (cycle = 1, 2, …, *MCN*), where *MCN* is the maximum cycle number of the search process. Then, an employed bee modifies the position in her memory depending on the local information (visual information) and tests the nectar amount of the new position (modified solution). If the nectar amount is higher than that of the previous one, the bee memorizes the new position and forgets the old one. Otherwise, she keeps the position of the previous one in her memory. After all employed bees have completed the search process; they share the nectar information and their position with the onlooker bees waiting in the dance area. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with a probability related to its nectar amount. The same procedure of position modification and selection criterion used by the employed bees is applied to onlooker bees. The greedy-selection process is suitable for unconstrained optimization problems. However, to overcome the greedy-selection limitations, specifically in a constrained optimization problem [82], the Deb's constraint-handling (feasibility-based) method [27] is adopted. The probability p_i of selecting a food source by onlooker bees is calculated as follows:

$$p_i = \frac{fitness_i}{\sum_{i=1}^{E_b} fitness_i} \tag{3.1}$$

where, $fitness_i$ is the fitness value of a solution i, and E_b is the total number of food-source positions or, in other words, half of the *CS*. Clearly, resulting from using (3.1), a good food source will attract more onlooker bees than a bad one. Subsequent to onlookers selecting their preferred food source, they produce a neighbour food-source position $i + 1$ to the selected one i, and compare the nectar amount of that neighbour $i + 1$ position with the old i position. The same selection criterion used by the employed bees is applied to onlooker bees. This sequence is repeated until all onlookers are distributed. Furthermore, if the i^{th} solution does not improve for a specified number of times (*limit*), the employed bee associated with this solution abandons it, and she becomes a scout and searches for a new random food source. Once the new position is determined, another ABC algorithm cycle (*MCN*) starts. The same procedures are repeated until the stopping criteria are met.

In order to determine a neighbouring food-source position, the ABC algorithm alters one randomly chosen parameter and keeps the remaining parameters unchanged. In other words, by adding to the current chosen parameter value the product of the uniform variant [-1,1] and the difference between the chosen parameter value and other "random" solution parameter values, the neighbour food-source position is created. The following expression verifies that:

$$x_{ij}^{new} = x_{ij}^{old} + u\left(x_{ij}^{old} - x_{kj}\right) \tag{3.2}$$

where, $k \neq i$ and both are $\in \{1, 2, \ldots, E_b\}$. The multiplier u is a random number between [-1,1] and $j \in \{1, 2, \ldots, D\}$. In other words, x_{ij} is the j^{th} parameter of a solution x_i that was selected to be modified. When the food-source position has been abandoned, the employed bee associated with it becomes a scout. The scout produces a completely new food-source position as follows:

$$x_i^{j\,(new)} = min\, x_i^j + u\left(max\, x_i^j - min\, x_i^j\right) \tag{3.3}$$

where, (3.3) applies to all j parameters, and u is a random number between [-1,1]. If a parameter value produced using (3.2) and/or (3.3) exceeds its predetermined limit, the parameter can be set to an acceptable value [82].

Clearly, employed and onlooker bees select new food sources in the neighbourhood of the previous one depending on visual information based on the comparison of food-source positions [84]. On the other hand, scout bees, without any guidance looking for a food-source position, explore a completely new one. Scouts are characterized, based on their behaviour, by low search costs and a low average food-source quality. Occasionally, the scouts can fortunately discover rich, entirely unknown food sources. In the case of artificial bees, the artificial scouts could have the fast discovery of feasible solutions as a task [87]. The flowchart of the ABC algorithm is exemplified in Figure 3.1.

Parameter-tuning, in meta-heuristic optimization algorithms, influences the performance of the algorithm significantly. Divergence, becoming trapped in local extrema, and time-consumption are consequences of improper parameters setting. An advantage of the ABC algorithm is having few controlled parameters. The ABC algorithm does not depend on the initial population because initializing it "randomly" with a feasible region is sometimes cumbersome. Instead, its performance sufficiently directs the population to the feasible region [85].

The controlled parameter (*limit*) is important because it prevents the algorithm from getting trapped in a local extrema. Therefore, it was suggested in [82, 88, 89] and also proven throughout this survey, that the optimal adjustment of the *limit* parameter is 50% of the product of *CS* and *D*.

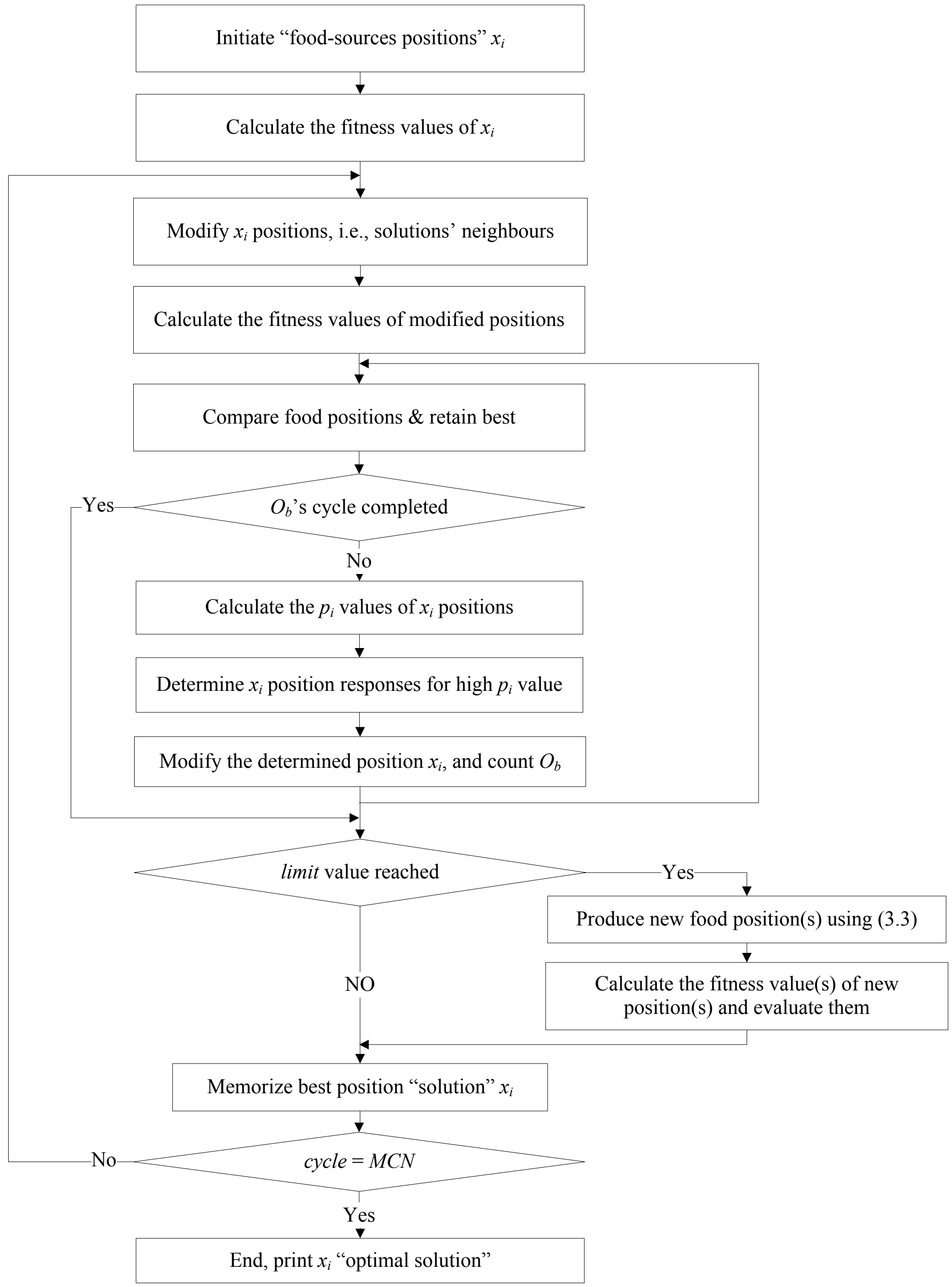


Figure 3. 1: Flowchart of the ABC algorithm.

It is clear that the ABC algorithm has the following control parameters: 1) the *CS* that consists of employed bees (E_b) plus onlooker bees (O_b), 2) the *limit* value, which is the number of trials for a food-source position to be abandoned, and 3) the maximum cycle number (*MCN*). Although the ABC algorithm has three parameters to be tuned, once the *CS* parameter has been determined by the practitioner, the *limit* value can be calculated easily. Therefore, technically speaking, the ABC algorithm has only two parameters to be adjusted: *CS* and the *MCN* values. Updating these two parameters towards the most effective values has a higher likelihood of success than in other competing meta-heuristic methods. The pseudo-code of the ABC algorithm is as follows:

1. Initialize the population.
2. Modify positions.
3. Apply selection criterion.
4. ***Repeat*** (cycle).
5. Allow the employed bees to share the food information with onlooker bees.
6. Allow the onlooker bees to choose the best food source based on the probability calculation.
7. Apply selection criterion.
8. Check for an abundant solution, and (if exists) initiate a new food-source position. Otherwise, follow the next step.
9. Retain best solution so far.
10. ***Until*** stopping rule.

3.3 A LITERATURE REVIEW

Decades ago, researchers [90-94] were motivated by the intelligent behaviour of honey bees. Bee swarming was adopted [95-98] to solve optimization problems and showed promising results. In addition, the bee swarm tactic has been employed since the early 1990s in a wide range of areas and applications as presented in [99].

Although the ABC algorithm was only recently introduced, the trend of published papers utilizing this algorithm, as illustrated in Figure 3.2, is growing rapidly. Furthermore, the performance of the ABC algorithm, and the results and quality of the solutions, outperformed or matched those obtained using other well-known optimization algorithms. The following sub-sections categorize the areas utilizing the ABC algorithm. The distribution of ABC's literature among a wide area is demonstrated in Figure 3.3. Table 3.1 (in pages 51-53) provides a summary of each paper utilizing the ABC algorithm in terms of the number of tested problems and the competing algorithms.

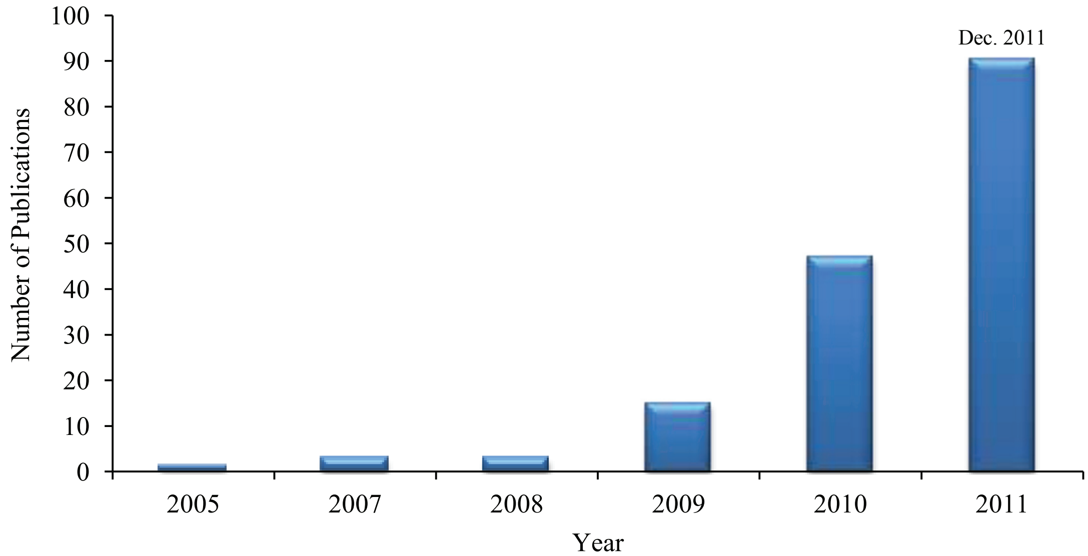


Figure 3. 2: Number of ABC related publications per year.

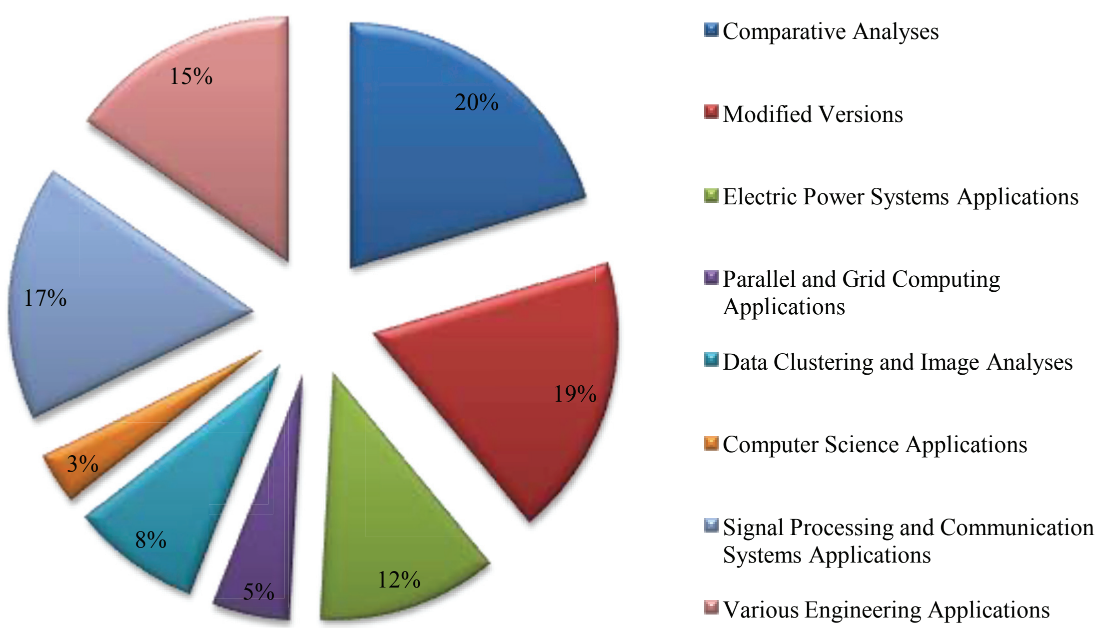


Figure 3. 3: Division of published ABC's papers due to different applications.

3.3.1 COMPARATIVE ANALYSES

Karaboga *et al.* in [85] and [100] extended the ABC algorithm to handle constrained optimization problems. A set of 13 benchmark optimization problems was examined, and the results were compared with state-of-the-art unconventional optimization algorithms in both references. However, the performance of most meta-heuristic optimization algorithms is independent of constraints handling methods. In other words, a superior optimization algorithm would diverge or be trapped in local extrema if the constraints were handled inefficiently. The same authors of [85] in [84] examined the performance of the ABC algorithm in five benchmark optimization functions, with the dimension of each function varying from 10 to 30. The results outperformed other meta-heuristic and hybrid algorithms such as PSO, GA, and hybrid evolution algorithm (EA) and PSO (EA-PSO).

The authors of [88] performed an analysis of the control parameters of ABC algorithm with a set of five benchmark optimization functions and compared the results with those of PSO, EA, and DE algorithms. Furthermore, three of the functions examined have 50 parameters to be optimized. Tuning the *limit* parameter (scout bee cycle) to be half the colony size multiplied by the problem's dimension resulted in best ABC performance – especially in highly nonlinear and non-convex optimization functions.

A large set of fifty benchmark optimization functions was solved by the ABC algorithm in [89]. The results were compared with those obtained using other competing methods in three experiments. The first one included GA, PSO, and DE algorithms, and the second contained various types of evolution strategy (ES) algorithms. Additional ES algorithms have been considered in the third experiment, as well. Throughout this large comparative analysis, statistical measurements proved the superiority of the ABC algorithm compared to most other well-known meta-heuristic optimization algorithms.

The authors of [101] utilized the ABC algorithm in solving large-scale optimization problems. Two tests, i.e., nine benchmark, unconstrained, optimization functions and five engineering, constrained, optimization problems were adopted to illustrate the performance of the ABC method. In addition, the results of the proposed algorithm were compared with those attained using various meta-heuristic techniques. The superiority of the ABC algorithm in handling both constrained and unconstrained optimization problems was demonstrated. Furthermore, the constraints-handling method utilized in their paper was based on the feasibility of a solution.

The Sudoku puzzle was solved in [102] via the ABC algorithm. The puzzle is considered a logic-based problem with three constraints. The first and second conditions are that each row and column must have 1-9 digits with no repetition. In addition, the third and dominant constraint indicates that every three-by-three block consists of non-repeated 1-9 digits. The authors considered three types (easy, medium, and hard) of Sudoku puzzles to demonstrate the efficiency of their proposed method. The offered method outperformed other GA-based Sudoku solutions. Slight modifications were applied to the ABC algorithm to suit the problem considered, e.g., onlooker bees were tuned to be larger than the employed bees and scout bees were adjusted to 10% of employed bees.

Prediction of the protein structure represents a challenging biochemistry energy minimization task which can be solved experimentally or computationally [103]. Experimental results are accurate but are expensive and time consuming, i.e., it takes months to verify the protein structure [104]. Due to these limitations, scientists solved this problem as an optimization problem aiming to find the lowest energy confirmation. Solving the protein confirmation problem led to several advantages, e.g., creating new medicine and analyzing genetic diseases [105]. The authors of [103] and [104] utilized the ABC algorithm to identify the protein structure, and compared the results to other techniques. The outcomes of ABC algorithm outperformed those attained by the parallel SA-GA methods.

In [106] the ABC algorithm was applied to solve integer programming problems, where all optimized parameters have to be integers. The integer characteristics of the problems were basically handled by rounding off the calculated solutions to the closest feasible values. Several test problems considered, and the results compared with those obtained utilizing a variety of PSO techniques.

Akay and Karaboga, in [107] analyzed the effect of parameters-tuning on the performance of the ABC algorithms. A set of nine benchmark optimization functions were adopted and the results were compared with those of PSO and DE techniques. The dimensions of the tested functions were 10, 100, and 10^3 parameters. As well as the effect of the colony size, feasible region scale, and *limit* value were investigated. The control parameter (*limit*) played a key role in the ABC performance. In other words, adjusting the *limit* parameter to a small value in a relatively small population led to an exploration process that overrode the exploitation routine.

The authors of [108] utilized the ABC algorithm to train feed-forward neural networks. The objective was to minimize the network error via varying weights. Three tests were considered: Exclusive-OR, 3-Bit Parity, and 4-Bit Encoder-Decoder problems. The solutions obtained by the proposed algorithm were compared and outperformed those found using the GA and back propagation (BP) techniques.

3.3.2 MODIFIED VERSIONS

A modification to the basic ABC algorithm, claimed in [109], occurred during the production of a neighbouring solution by introducing additional parameters to the original algorithm. Instead of changing one parameter in a solution vector, the modification rate (*MR*) was suggested to control the number of altered parameters in a solution vector. In addition, a scaling factor (*SF*) was proposed to replace the uniformly distribution multiplier (*u*). The *SF* (step size) could be utilized (adaptively) dynamically to enhance the algorithm's performance contrary to a fixed random number between [-1,1] used in the basic version. Two sets of benchmark optimization functions were designated to confirm the effectiveness of the modified version. The outcomes of these experiments compared with those of other (competent) well-known optimization methods. The authors demonstrated the superiority of both the basic and modified ABC algorithms in solving various types of optimization applications. However, adding more parameters to be tuned decreased the likelihood of success in updating these parameters to the optimal adjustment.

A modified version of the ABC algorithm was proposed in [110]. The main difference in this paper was that once a solution *i* did not improve for a specified number of trials, the whole algorithm was terminated. Subsequently, the employed bees became scouts and explored new solutions randomly. Despite that, such termination of an algorithm increased the diversity of an exploration process over the exploitation process. The authors considered a set of 10 benchmark optimization functions, with few optimized parameters. The results were not compared with any algorithm.

Instead of the initial random solutions used in the original ABC algorithm, the author of [111] proposed three versions based on chaotic mapping sequences. The author stated that “a chaotic map is a discrete-time dynamical system running in chaotic state.” Therefore, the chaotic sequence can be recognized as a random [0,1] number sequence. The adopted chaotic functions were Logistic, Circle, Gauss, Henon, Sinusoidal, Sinus, and Tent maps. The first (C1-ABC) and second (C2-ABC) modified versions were concerned with changing the calculations of initial-phase and scout-phase utilizing the selected chaotic maps. A combination of those two versions represented the third offered type (C3-ABC). The performance was assessed via three benchmark optimization functions, and was compared with the original ABC method. The results obtained by the proposed versions outperformed the original ABC algorithms. However, C2-ABC and C3-ABC attained better outcomes than the C1-ABC technique.

In order to improve the exploitation strategy of the ABC algorithm, the authors of [112] integrated the global population’s best value in the solution procedures. In other words, they adopted the *G-best* term used in PSO algorithm to modify (update) the solutions in the ABC method. Six experimental tests were utilized to reveal the efficiency of the proposed modified version. Slight enhancements were obtained in some of the designated functions due to this modification. However, an additional parameter to be tuned, i.e., a *C* multiplier, was a drawback of such an adjustment.

Cooperative strategies to the ABC algorithm were proposed in [113] and [114], inspired by the idea of retaining the global best solution and utilizing its components. This routine has been used extensively in the PSO and its derivatives’ algorithms. However, such a modification will alter the balanced mechanism of the original ABC algorithm, affecting its performance in some problems. In addition, retaining the best solution’s vector at every population per cycle would lead to a significant requirement of the CPU time. As reported in [113], the results of the original ABC method outperformed other competing (cooperative) approaches at two of the six designated benchmark functions. A hybrid technique was also presented in [114] using the original ABC algorithm in sequence with the suggested cooperative method. In other words, the offered cooperative practice (initially) optimized each parameter of a solution vector independently. Subsequently, this information exchanged with the original ABC algorithm in the next iteration, and so forth. The author of [114] analyzed the performance of the suggested approach via seven frequently tested optimization functions. Once more, slight improvements were obtained by the proposed approach in only two of the seven selected functions.

The authors of [115] integrated an exponential adaptive scaling factor with the original ABC algorithm that was activated during the exploitation practice. The motivation was to cover a wide range of the search-space and to avoid being trapped in a local solution. However, additional parameters to be tuned were included. Two sets of optimization functions were adopted to prove the efficiency of the offered adaptation. However, the performance of the adaptive ABC algorithm deteriorated when the problem's dimensions increased.

An elitist ABC algorithm was proposed in [116]. The authors suggested five modification or enhancement steps to the original ABC method. First, the step size of modifying a solution was doubled to expand the search-space covered. Second, each onlooker (per iteration) searched nearby the best found solution so far, and the step size of the onlooker cycle was doubled. Third, the scout bee was generated in the same fashion of solution's modification used in PSO algorithm. Fourth, a dynamic tolerance factor was superimposed to fulfill the equality constraints of a problem. Fifth, two local search mechanisms were activated at specific times to enhance the exploration of the retained solution. Clearly, several additional parameters were considered in this approached requiring considerable time in the trial-and-error experiments. A set of 18 optimization problems was selected to exemplify the competence of this approach. Negative impact on the solution quality, however, was reported when the problem's dimensions were larger than 30. In addition, the results were not compared with any algorithm.

Arguing that the original ABC algorithm has a slow convergence speed, the authors in [117] offered a modified ABC version. First, the entire population shared the best "so-far" position assuming that the global solution closed to that position. This assumption might be invalid, specifically at a non-convex search-space. Second, they altered the step size (multiplier used to generate a new solution) from the static to a dynamic mechanism, i.e., indirect relation to the number of iterations. Third, the greedy selection of the best solution was changed to be only in favor of objective function instead of fitness value. A set of benchmark optimization functions and image registration applications was utilized to evaluate the effectiveness of the suggested version. The results of the modified ABC approach outperformed those of the original algorithm at fixed initial positions and desired mean values. However, when these restrictions were relaxed (particularly) at the image experiments, the enhancements of the proposed version were insignificant.

Replacing the scout behaviour by a greedy-selection practice with a mutation process was the modification imposed on the original ABC algorithm in [118]. These two modifications shared the information of the entire population to search for the potential region. The authors used a set of six benchmark optimization functions with a global solution at the origin to validate the suggested method's performance, where most existing optimization techniques performed extremely well for such functions. Practical optimization problems rarely have a global solution at the origin of the search space. In comparing with the original ABC version, slight improvements in the obtained results were due to the implementation of the proposed method.

3.3.3 ELECTRIC POWER SYSTEMS APPLICATIONS

The economic dispatch problem was solved in [119] using the ABC algorithm. The objective was to minimize the total system fuel cost subject to operational equality and inequality constraints. Three test systems were used to evaluate the performance of the algorithm. In addition, the results of the ABC algorithm were compared with deterministic (lambda) technique and various types of PSO and GA algorithms. The results obtained using the ABC algorithm outperformed those competing methods.

The authors of [120] and [121] used the ABC algorithm to obtain the optimal DG size, location, and power factor in distribution feeder systems. The objective was to minimize the total system power losses subject to equality and inequality constraints. Furthermore, the performance of the ABC algorithm with discretized parameters was highlighted. The authors of [120] proposed a modified version enhanced the solution quality and accelerated the algorithm performance. This modification occurred during the neighbouring search process specifically in the onlooker routine. The IEEE 33-bus and 69-bus radial distribution feeder systems were considered, and the results of the offered method compared with the original ABC method and the exhaustive (grid-search) approach. Moreover, the same authors employed the ABC algorithm to assess the priority-ordered constrained search technique for distributed generation application problems offered in [122].

The authors of [123] and [124] utilized the ABC algorithm for distribution networks reconfiguration applications. The objective was to minimize the total system power losses subject to equality constraints, i.e., voltages and thermal capacity limits. The desired solutions were attained based on reconfiguring the switches on the feeder systems. The 14-bus, 33-bus, and 118-bus radial distribution feeder systems were designated for testing the algorithm performance. In addition, the results compared with SA, TS, DE, and GA methods. The outcomes attained by the ABC method were virtually identical to those compared algorithms. However, significant reductions in the required CPU time were due to the utilization of the ABC method.

In [125] the OPF problem was solved using the ABC algorithm. The objective was to minimize the system’s total operating fuel costs adopting the quadratic fuel formula. The equality constraints were the real and reactive power balance equations. Security inequality constraints were considered in this problem. The authors adopted the IEEE 14-bus and IEEE 30-bus systems, and the results compared with those obtained using the GA and PSO methods. In both cases, the solutions of ABC algorithm outperformed those of the compared methods in terms of solution quality and algorithm's performance.

3.3.4 PARALLEL AND GRID COMPUTING APPLICATIONS

Narasimhan in [126] proposed a parallel ABC algorithm to solve numerical optimization functions. First, the colony of bees was distributed equally at each designated processor. Then, solutions obtained from each processor were recorded in a local memory. After that, a global-shared memory retained the improved solutions that were attained from each processor, and utilized them to generate the neighbouring solutions per colony set. Eventually, the last updated solution recorded in the shared memory contained the optimal result. The author used six benchmark optimization functions to assess the performance of the proposed parallel ABC algorithm with respect to the original algorithm.

The authors of [127] used the ABC algorithm in a parallel computing environment, i.e., via four machines distributed as a manager processor and three subsequent ones. Although each processor performed the ABC algorithm routine, an exchange of two randomly selected solutions occurred before the next iteration. The exchange controlled by the manager processor by offsetting the worst performing population of the subsequent processors. A set of six benchmark optimization functions was designated to demonstrate the efficiency of the suggested method. The results of the approach showed an improvement in solution quality. Furthermore, a dramatic reduction in the required CPU time was illustrated when the number of processors increased.

Three benchmark functions were utilized to show the efficiency of the proposed parallel ABC algorithm in [128]. Basically, the authors divided the ABC strategy into several groups (subpopulations) that performed independently. After a predetermined number of iterations, those groups exchanged the obtained information. In other words, the population with better fitness values will take over those with worse fitness values. A set of three benchmark functions was adopted to verify the efficiency of the proposed method. The outcomes outperformed or matched those obtained using other algorithms. Such a strategy led to insignificant enhancements with respect to solution quality and required CPU time. However, the additional parameter to initiate the exchange task decreased the likelihood of success in updating the overall algorithm's parameters.

3.3.5 Data Clustering and Image Analyses

The ABC algorithm was utilized in [129] to detect the image of an aircraft at low altitude. In addition, an attractive pattern scheme was utilized in the optimization process by maximizing the similarities between the target and actual images. The performance of the ABC algorithm outperformed the GA method in all the considered test cases. The authors highlighted additional insights of the convergence and complexity of the ABC algorithm through statistical records.

Ref. [130] used the ABC algorithm for image detection problems. Four experiments were conducted to find the pattern of an object with grey and coloured images. In addition, the results of the offered method outperformed those of GA and PSO techniques. The Euclidean distance formula was used in [131] to solve the clustering problem via the ABC algorithm, adopting the feasibility selection criterion instead of the frequently used greedy-selection mechanism. The performance of the proposed algorithm was assessed after carrying out a set of three benchmark clustering problems. A variety of commonly used statistical methods was adopted for comparison.

The ABC algorithm in [132] was used for clustering analysis, i.e., to cluster (x) objects into (y) clusters. Basically, the objective was to minimize the Euclidean distance between an object and its associated cluster centre. A set of 13 benchmark clustering problems was tested, and the PSO with nine other classical techniques adopted for comparison purposes. Simulation results showed that the performance of the ABC algorithm outperformed those competent techniques.

The authors of [133] utilized the ABC algorithm in solving the data clustering problem. In addition, they modified the original ABC method to enhance the ABC's performance, specifically during the position update routine. However, these modifications were adopted from the PSO technique resulting in additional parameters to be adjusted. The authors initially evaluated the modified version by solving different benchmark optimization functions. Then, they used the proposed algorithm to solve two data clustering problems. Based on the comparison analysis, the results of the suggested version outperformed those attained by other approaches.

3.3.6 COMPUTER SCIENCE APPLICATIONS

The leaf-constrained minimum spanning tree (LCMST) problem has been solved in [134] through the ABC algorithm. The objective was to find a spanning tree contained specific number of levels with a minimum total weight. The LCMST utilized in practical applications such as facilities location and network designs [135]. In addition, the results of the algorithm via large-set problems outperformed those obtained using other commonly used methods such as GA, TS, and ACO techniques.

To maximize financial profit, a hybrid wavelet transform (WT), recurrent neural networks (RNN), and ABC algorithms were utilized in [136] for the stock market's price forecasting. The choice of RNN scheme was due to its merits in solving complex time series' applications in comparison with feed-forward networks. The neurons within the multiple hidden layers of RNN were connected to themselves and to the adjacent ones as well. Each algorithm has a dedicated assignment. In other words, the WT used to eliminate the noise of the data sample; adjusting the inputs' values to match the desired outputs was assigned for RNN method; optimal tuning of the RNN's weights was designated for the ABC algorithm. Various international stock markets' indices were adopted to evaluate the performance of the suggested price forecasting tool. The outcomes of the proposed tool during the selected six-year periods outperformed those attained based on fuzzy and NN approaches. However, the fuzzy based approach did better than the offered tool in 2002 and 2003.

3.3.7 SIGNAL PROCESSING AND COMMUNICATION SYSTEMS APPLICATIONS

The author of [87] used the ABC algorithm in designing the digital infinite impulse response (IIR) filters to minimize the error between the desired and unknown output signals of the system. The order of the considered filters' transfer functions was 1st, 2nd, 4th, and 20th via four experiments. Moreover, the results of the ABC algorithm were compared with those of the least square error (LSQ) and PSO methods and the impact of the ABC parameters on the algorithm's performance was investigated. The author also evaluated the performance of the ABC algorithm over four benchmark optimization functions. Increasing the filter's order (up to 34^{th}) was presented in [137] and solved by the ABC algorithm. The outcomes of the ABC algorithm, in [87] and [137], outperformed those attained using competing techniques.

Ref. [138] used an ABC algorithm in solving applications of the wireless sensory network (WSN) problems. The objective was to maximize the coverage area and minimize the number of sensor nodes. The optimization problem was constructed in a clustering problem fashion, i.e., for a set of locations and a specific number of sensors, the optimal location to utilize all sensors was obtained so that every location was covered with minimum sensors. Three experiments with a variable number of sensors were conducted, and the results were identical to those attained using an analytical approach.

Sabat *et al.* [139] used the ABC algorithm for the metal extended semiconductor field effect transistor (MESFET) applications. The objective was to design a MESFET device to minimize the error (difference) between the desired and calculated parameters. In addition, the design problem was solved by the proposed ABC algorithm and the PSO technique. The results of the proposed algorithm outperformed those obtained using the competing method by means of solution quality and less computing time.

Bernardino *et al.* [140] solved the non-split weighted ring loading problem (non-split WRLP) using the ABC algorithm. The aim of a non-split WRLP problem was to minimize the weights (demands) of a transmission routing (path) in a network for a predetermined set of communication orders. The authors modified some of the ABC algorithm features by using deterministic methods to determine the initial (discrete) population, rather than the random, solutions. Moreover, they analyzed the effect of varying the ABC's parameters on the algorithm performance and considered a set of test examples with different ring sizes and demand values. The results of the proposed ABC algorithm outperformed those obtained using other well-known methods – in particular, when the dimension of the problem was increased.

The authors of [141] utilized the ABC algorithm to minimize the transient performance, i.e., propagation delay time, of a complementary metal oxide semiconductor (CMOS) inverter. Three inverter-designed cases were used to examine the ABC method. The outcomes showed that the ABC algorithm was significantly faster than the compared SPICE (tool) simulation. Due to the positive merits of ABC algorithm with respect to traditional design tools, further investigation of designing digital microelectronic circuits by the ABC algorithm was suggested in [141].

The gains of a proportional integral derivative (PID) controller were optimized using the ABC algorithm in [142]. The objective was to minimize the design error, overshoot time, and settling time. Three test controllers with high orders were utilized to show the effectiveness of the algorithm. The outcomes of the ABC method outperformed those found by other well-known techniques. The dynamic response of a proportional integral (PI) controller was optimized by a combination of GA and ABC algorithms in [143]. The hybrid method utilized the GA initially and then the ABC routines. This sequence repeated upon meeting the stopping criteria, i.e., maximum iteration reached or design specifications met. The gains' values were optimized due to the specified design objectives, i.e., minimum overshoot time, rising time, settling time, and steady state error. The experimental results proved that the hybrid GA-ABC algorithm scored the best gains with respect to the compared methods.

The authors of [144] used the ABC algorithm to solve the WSN problem in a clustering mode. The objective was to minimize the energy consumed by each sensor and maximize the network's lifetime. In this paper, the network was consisted of a cluster-head and its (clusters) members during predetermined periods. The main goal of the cluster-head was to collect the information from its members and send it to the base. Therefore, the cluster-head consumed greater energy than its members. Accordingly, minimizing the energy dissipation was the aim of the cluster style considered. Two well-known techniques used to solve the WSN problems were implemented for comparison purposes. The results of using the ABC algorithm outperformed those techniques, hence increasing the network's lifetime.

A comparison between the GA and ABC algorithms' performances was conducted in [145] to solve the localization problem of WSN. The aim was to minimize the distance between the measured and reference points within the designated environment. Several obstacles were embedded in that environment (search-space) for additional challenges to assess the performance of the considered algorithms. The outcomes of ABC algorithm outperformed those obtained by the GA method significantly by fewer location errors and shorter consumption time.

3.3.8 VARIOUS ENGINEERING APPLICATIONS

The authors of [146] proposed a discrete version of the ABC algorithm (*DABC*) to solve the lot-streaming flow shop scheduling problem. The main idea (objective) was to minimize the job's completion time by allowing overlaps between successive machines and their operation tasks. To evaluate the proposed algorithm outcomes, two types of the flow shop scheduling problems under a set of 20 problems have been examined. Statistical measurements were employed to show that the *DABC* algorithm produced improved results in contrast with those attained using other hybrid meta-heuristic (PSO-based and GA-based) algorithms. Another version of the flow shop scheduling problem solved by the same authors in [147, 148] using their *DABC* algorithm introduced in [146]. The permutation flow shop scheduling problem (PFSP) can be interpreted as allocating each task (job and corresponding operations) associated with every machine. However, each machine restricted to perform a single operation only without interruption or time-delay in the production line. In other words, the objective, in [147] and [148], was to minimize the total flow time citation. A set of 90 benchmark scheduling problems was considered for comparison reasons. The suggested algorithm outperformed 71% and 49% of the considered problems reported in [147] and [148] respectively. Moreover, minimizing the production time subjected to different inequality constraints was the aim in [149]. The authors used the ABC algorithm to solve the multi-pass milling operation problem. A range of cutting strategies, e.g., speed and depth of the cut, was adopted towards the most optimal solution and a detailed comparison of the results was presented.

From a supply chain management perspective, the authors of [150] adopted the ABC algorithm in the problem considered. The objective was to minimize the remanufacturing costs subject to various equality and inequality constraints representing the behaviours of facilities and end users. The results of ABC algorithm outperformed those of the PSO method by a significant reduction in the desired function.

An optimal structure design problem, specifically minimizing the weight of the structural design, was solved using the ABC algorithm in [151]. Various inequality constraints based on the loading conditions were taken into account in the optimization process. The author converted this constrained problem to an unconstrained one by a dynamic penalty formula. A set of five frequently tested design problems was utilized to evaluate the effectiveness of the proposed technique. From the comparison analysis, the ABC's outcomes substantially outperformed those attained by other well-known methods. In addition to minimizing the structural's weight, the authors of [152] employed the ABC algorithm to solve a multi-objective optimization problem. The bi-objectives were minimizing the weight and cost of the composite structure, subject to strength constraints. Three design problems were conducted, and the results of the proposed ABC method outperformed those competing algorithms.

The authors of [153] used the ABC algorithm to solve the mechanical draft cooling tower problem. Most large thermal industries (systems) use these cooling towers to dissipate the process heat. The objective was to minimize the total (operational and capital) costs subject to satisfying various design's inequality constraints. Frequently used examples were adopted to show the efficiency of the proposed algorithm. A high-level mathematical tool [154], general algebraic modeling system (GAMS), was designated for comparison. In all the tested cases, the outcomes of ABC algorithm significantly outperformed those of GAMS, saving thousands of dollars.

The ABC algorithm utilized to solve mixed integer nonlinear optimization problems in [155]. The objective was to maximize the system's reliability subject to complex design's constraints. Different types of design systems (series, parallel, and combination of series and parallel) were adopted to reveal the performance of the ABC algorithm. The outcomes of the ABC method, in all test systems, showed the highest reliable results.

Table 3. 1: Summary of published papers utilizing the ABC algorithm w.r.t. experimental tests and compared algorithm; --: no comparisons were carried out.

Ref.	Tests	Compared Algorithms			
[82]	3	--			
[84]	5	PSO	GA	EA-PSO	
[85]	13	PSO	DE		
[87]	8	PSO	LSQ		
[88]	5	PSO	DE	EA	
[89]	50	GA	PSO	DE	ES
[99]	Survey	--			
[100]	13	Various Stochastic-based Algorithms	PSO	DE	GA
[101]	14	Various Heuristic Techniques			
[102]	3	GA			
[103]	4	Different Heuristic Algorithms			
[104]	1	Parallel SA-GA Methods			
[106]	9	Different PSO Techniques			
[107]	9	PSO	DE		
[108]	3	Back Propagation	GA		
[109]	33	Different PSO Methods	ES	GA	DE
[110]	10	--			
[111]	3	ABC	C1-ABC	C2-ABC	C3-ABC
[112]	6	ABC			
[113]	6	ABC	PSO		
[114]	7	ABC			
[115]	11	ABC	GA	Various PSO Methods	
[116]	18	--			
[117]	10	ABC			
[118]	6	ABC			

Ref.	Tests	Compared Algorithms				
[119]	3	Different PSO Methods	Lambda	GA		
[120]	12	ABC	Grid-Search	Analytical		
[121]	9	Grid-Search	Analytical	GA		
[122]	4	Nonlinear Optimization	Grid-Search	Analytical	ABC	GA
[123]	3	SA	TS	DE	GA	
[124]	2	SA	DE			
[125]	2	GA	PSO			
[126]	6	ABC				
[127]	6	ABC				
[128]	3	ABC	PSO			
[129]	4	GA				
[130]	4	PSO	GA			
[131]	3	Hybrid PSO-based Methods	TS	SA	ACO	GA
[132]	13	9-Classical Clustering Methods	PSO			
[133]	7	ABC	Various PSO Algorithms			
[134]	65	GA	TS	ACO		
[136]	8	ANN and Fuzzy based Methods				
[137]	4	PSO	GA			
[138]	3	Analytical				
[139]	1	PSO				
[140]	19	GA	TS	PSO		
[141]	3	SPICE Simulation				
[142]	3	Various Deterministic Methods	Fuzzy-based	GA	EP	
[143]	1	ABC	GA	Gradient		
[144]	1	Different Methods used for WNS Applications				
[145]	6	GA				

Ref.	Tests	Compared Algorithms			
[146]	40	Hybrid GA and PSO based Algorithms			
[147][148]	90	Hybrid Heuristic-based Algorithms			
[149]	1	PSO	SA		
[150]	1	PSO			
[151]	5	Different Heuristic Algorithms			
[152]	3	GA	PSO	AIS	
[153]	7	GAMS Tool			
[155]	4	Fuzzy-based	DP	GA	AIS

3.4 Summary

The aim of this chapter was to address state-of-the-art research on the ABC algorithm. The algorithm was discussed, and the key features highlighted. Throughout the literature review, it outperformed or matched other well-known meta-heuristic algorithms.

Furthermore, parameter-tuning in meta-heuristic optimization algorithms significantly influenced the algorithm's performance. Unlike the ABC algorithm, with only two parameters (*CS* and *MCN*) to adjust, other well-known meta-heuristic algorithms have many parameters to tune. Updating the two parameters towards the most effective values has a higher likelihood of success than other competing meta-heuristic algorithms. The performance of the ABC algorithm demonstrated its superiority and potential for solving complex real-world problems in future research.

CHAPTER 4: NOVEL META-HEURISTIC TECHNIQUE: SENSORY-DEPRIVED OPTIMIZATION ALGORITHM

4.1 INTRODUCTION

Most efficient meta-heuristic algorithms are inspired by natural phenomena [79-83]; therefore other superior algorithms can also be derived from nature. The terms *heuristic* and *meta-heuristic* are typically used in the literature interchangeably, but the main difference between these two terms is as follows. A heuristic algorithm is set rules of thumb that lead to an optimal or quasi-optimal result. It does not require ideal data to achieve the desired solution; common sense approximations or assumptions drive the problem solution to attain superior results with varying degrees of certainties [156]. Therefore, heuristic algorithms are mostly designed for specific problems. On the other hand, a meta-heuristic algorithm is an advanced heuristic algorithm [83]. Meta-heuristic algorithms are generally not problem dependent, yet they follow the trial-and-error process and could combine more than one heuristic tactic to solve a problem [43, 62, 63].

Although meta-heuristic algorithms are random, they follow a logical pattern. Within most meta-heuristic algorithms, two procedures interact with each other. The first procedure explores the search space to arrive at the optimal region. Exploiting that region to determine an optimal solution is the goal of the second procedure. A good balance between these two procedures leads to a superior algorithm [86]. However, meta-heuristic algorithms have parameters to be adjusted, which vary from one application to another. An algorithm with few parameters to tune is always desired, as long as it is efficient.

Typically, the efficiency of meta-heuristic algorithms is confirmed by statistical measurements. Because obtaining the global solution is not guaranteed by meta-heuristic algorithms, they simulate independent runs to support their achievements. In addition, the required CPU time for an algorithm shares that evaluation, even though it mainly relies on the PC’s features, e.g., processor speed, random access memory (RAM) size, rather than the algorithm itself. Furthermore, the CPU time issue could be insignificant for off-line applications or planning-stage optimization problems. Optimization problems with large dimensions can be expected to consume additional time to avoid premature convergence.

In this chapter, a new sensory-deprived optimization algorithm (SDOA) is proposed. Similar to other meta-heuristic algorithms, the SDOA is a population-based and derivative-free. A set of benchmark optimization functions are examined to confirm the performance efficiency of the suggested algorithm. The results are compared with those found using other well-known meta-heuristic optimization algorithms.

4.2 THE SENSORY-DEPRIVED OPTIMIZATION ALGORITHM

Inspired by the intelligent behaviour or survival instincts of a sensory-deprived human being, this new meta-heuristic optimization algorithm is based on a solid concept and utilizes the exploration and exploitation processes simultaneously and distinctly from other meta-heuristic algorithms.

A sensory-deprived person (*SDP*) will use his/her functioning senses to help reach his/her goal. If a blind person wants to reach an object or to walk in a certain path, with neither guidance nor specific experience, he/she will rely intuitively on his/her remaining senses: tasting, touching, smelling, and hearing with various degrees of dependency. Only three of them – taste, touch, and smell – will provide him/her with feedback from nearby obstacles. The feedback from the remaining sense – hearing – could symbolize the slightly-distant obstacles. Utilizing these feedbacks will assist the blind person towards his/her goal.

It is clear that the exploitation and exploration processes are present in the above assumption. In other words, feedback from the taste, touch, and smell senses stand for the exploitation process of a region in the search space. However, for the exploration process, the routine is divided into two strategies. The first one employs hearing to receive feedback from an adjacent region of the search space of the remaining senses. The second (temporary) strategy terminates the worst-performing sensory-deprived persons (population) and replaces them by those who perform better in a parallel search process. This parallel search dynamically diminishes when the number of iterations increases. These procedures are repeated until the stopping criterion is met.

Initially, the populations adaptively update their standing positions using their available senses. Then, after a certain number of moving steps (R), individuals interact or communicate with each other to determine a leader. The leader is the one who has the highest probability of achieving the objective. Once the leader is determined, the remaining population follows him/her by searching nearby positions. After that, the leader and followers will be ranked based on their fitness functions' values. At the same time of selecting a leader, new rank positions are assigned, and evaluated. The number of those positions is defined by the user. This parallel process represents further exploitation action and dynamically decreases when the number of iterations increases. Afterward, the ranked solutions of both (main and parallel processes) are updated (swap or merge) if applicable, the best solution retained, and another iteration starts. These procedures are repeated a predetermined number of times.

The distinct and main advantage of the proposed algorithm is as follows. The feedback from the "hear" sense will be somewhere between the exploration and exploitation search, leading to covering a wide range of the solution search-space, and avoiding premature convergence. Furthermore, the solution vectors from the parallel semi-exploration routine enhance the solution vectors of the main algorithm. The flowchart in Figure 4.1 illustrates the solution procedures of the SDOA.

The initial solution (x_i) or standing position of a sensory-deprived person (*SDP*) is a *D*-dimensional vector with *D* parameters to be optimized and as follows:

$$x_i = [x_j \quad x_{j+1} \quad \cdots \quad x_D]_i \tag{4.1}$$

where, $i \in \{1, 2, \ldots, SDP\}$ and $j \in \{1, 2, \ldots, D\}$.

The procedures to modify the standing position of a *SDP* into four senses for the main search are as follows:

a) Randomly select the j^{th} parameter of an x_i solution to be modified.

b) Randomly retain the j^{th} parameter of an x_k solution, taking into account that $i \neq k$ and both i and $k \in \{1, 2, \ldots, SDP\}$.

c) Apply the following formulas:

$$x_{ij}{}^{new} = x_{ij}{}^{old} + u\left(x_{ij}{}^{old} - x_{kj}\right) \tag{4.2}$$

$$x_{ij}{}^{new} = x_{ij}{}^{old} + U\left(x_{ij}{}^{old} - x_{kj}\right) \tag{4.3}$$

d) Select the best standing position by means of fitness value.

Equation (4.2) adopted from [82] and utilized for the three senses searching nearby positions, and (4.3) used for the remaining (hearing) sense. The multipliers u and U are uniformly normally distributed random numbers, and their ranges have been chosen after conducting several experimental tests to various benchmark optimization functions. The multiplier in (4.2) is in the range [-1,1]. However, the U multiplier is in the range [-5,5] and excluding the interval of the u multiplier. Moreover, if the new parameter gets beyond the specified values it is adjusted to the nearest limit.

In the temporary search, steps a, b, and c are used but instead of selecting and retaining a single parameter, half of the D-parameters are selected and retained to be modified by this parallel search. Furthermore, the number of $SDP_{parallel}$ in this search is suggested to be ≤ 10% of the main population number. The parallel search starts with the main search and it gradually decreases when the number of iterations increases. In other words, assume that the main population is 100; therefore the parallel search population will be 10. Then, the population number of the parallel search becomes 9 in the second iteration, and so forth. The parallel search, in this example, will be terminated when the main search reaches the 11th iteration. Clearly, this procedure will enhance the premature solutions obtained by the SDOA. The probability of a solution i (p_i) is calculated based on the fitness values (fit_i), as follows:

$$p_i = \frac{fit_i}{\sum_{i=1}^{SDP} fit_i} \tag{4.4}$$

The solution procedures of the proposed SDOA including the *temporary parallel* search are as follows:

1. Initiate (randomly) standing positions (solutions) for the sensory-deprived individuals and evaluate the fitness values of those positions.
2. Modify the standing positions into 4 (senses) and evaluate them.
3. Compare modified positions with their standing positions and select the best new standing points.
4. If the populations update their standing positions R times, go to next step. Otherwise, repeat steps 2 and 3 R times (one time), where R is suggested as follows:

$$R \geq D^2 \tag{4.5}$$

5. A sensory-deprived individual associated with a standing position that represents the highest probability value will be selected as a leader.

5.1 If the main iteration number is larger than the population number of $SDP_{parallel}$, *go to step 6. Otherwise, go to the next step.*

5.2 Generate ≤ 10% of the main population, i.e., select random standing positions and evaluate them.

5.3 Modify the standing positions into 4 (senses) and evaluate them.

5.4 Update the $SDP_{parallel}$ *using* (4.6):

$$SDP_{parallel}^{new} = SDP_{parallel}^{old} - 1 \tag{4.6}$$

5.5 Rank the solutions in order.

5.6 Follow step 8.

6. Recruit the remaining population to search near the selected leader.
7. Evaluate the fitness values, and rank the solutions in order.
8. Swap or merge if possible.
9. Retain the best solution.
10. If stopping criterion reached, go to next step. Otherwise, follow step 2.
11. End, print solutions.

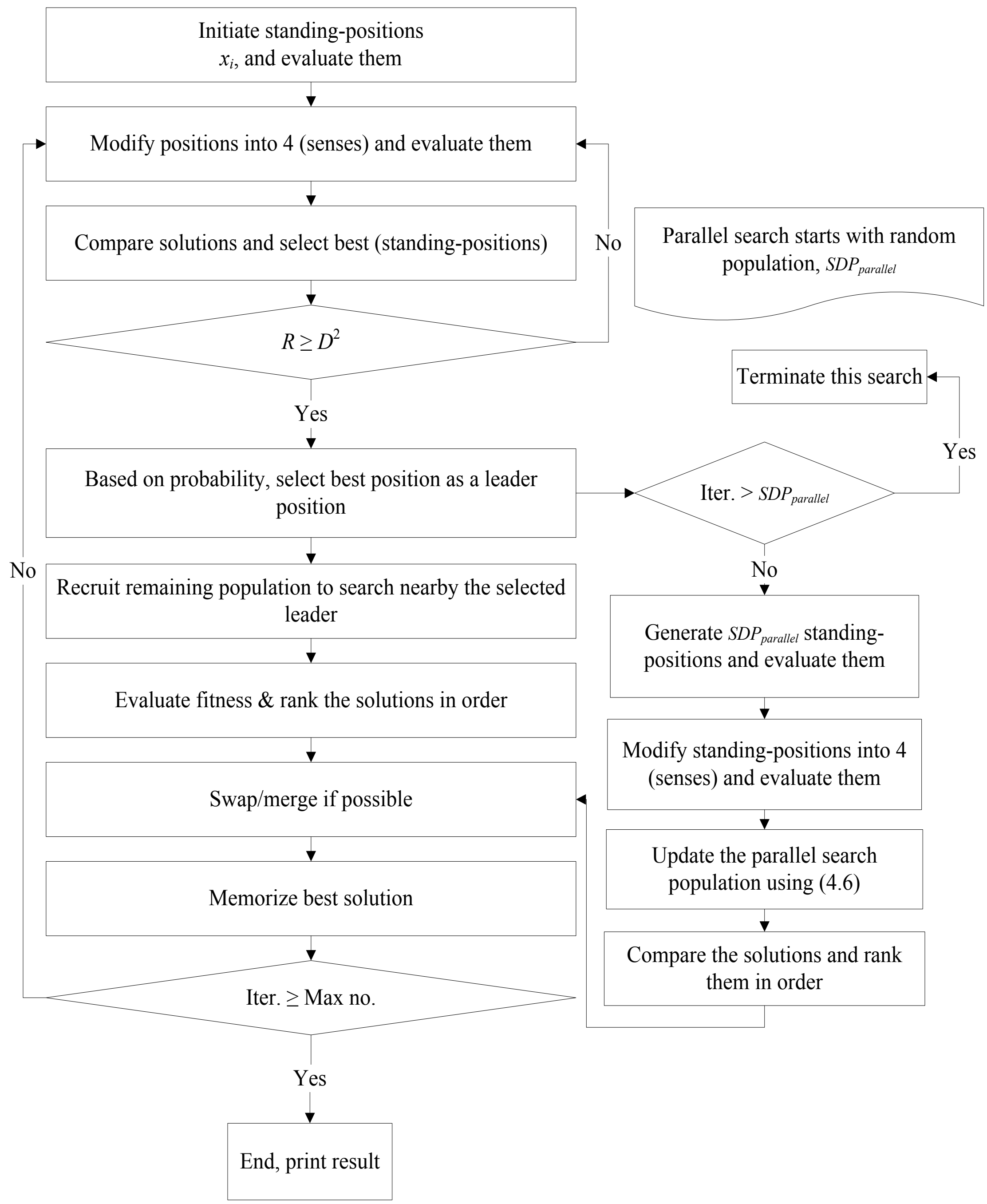


Figure 4. 1: Flowchart of the proposed SDOA.

4.3 EXPERIMENTAL RESULTS AND DISCUSSION

The absence of a mathematical framework of most meta-heuristic optimization algorithms makes it cumbersome to indicate which algorithm is better. Researchers attempt to verify the robustness of these algorithms using statistical measurements. In other words, benchmark optimization functions are typically utilized to show the efficiency of meta-heuristic algorithms. A set of benchmark functions used frequently in the literature is adopted in this chapter, as recorded in Table 4.1.

The results of the proposed algorithm after carrying out 30 independent runs were compared with those obtained using other well-known meta-heuristic algorithms, i.e., GA, PSO, ABC, and DE algorithms reported in [89]. In order to make the comparison realistic, the *SDP* number was tuned to 50, the maximum evaluation number was constrained to 5×10^5 and the values $\leq 10^{-12}$ were converted to zero as suggested in [89]. The results in terms of the mean and standard deviation (Std.Dev.) values are in Table 4.2. Although the authors in [89] neglected the required CPU time, it was calculated for the proposed SDOA as shown in Table 4.2 and Table 4.3.

Table 4. 1: Selected benchmark optimization functions; D: Dimension.

Function	x-Range	D	Formulation
f_1	[-100, 100]	30	$f_1(x) = \sum_{i=1}^{D} x_i^2$
f_2	[-30, 30]	30	$f_2(x) = \sum_{i=1}^{D-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$
f_3	[-100, 100]	2	$f_3(x) = 0.5 + \frac{sin^2\left(\sqrt{x_1^2 + x_2^2}\right) - 0.5}{\left(1 + 0.001(x_1^2 + x_2^2)\right)^2}$
f_4	[-600, 600]	30	$f_4(x) = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
f_5	[-5.12, 5.12]	30	$f_5(x) = \sum_{i=1}^{D}\left[x_i^2 - 10\, cos(2\pi x_i) + 10\right]$

The *Sphere* function (f_1), as Figure 4.2 shows, is a smooth problem with convex characteristics. It has no local minima except the global one at $f_1^* = 0$. However, the *Rosenbrock* function (f_2), known as the banana function, is a standard test function employed repeatedly. The global solution is inside a narrow and parabolic shape valley, as Figure 4.3 illustrates. This problem examines the algorithm's performance in avoiding being trapped in local minima. Although the *Schaffer* function (f_3) is a two-dimensional problem, the global solution for this function, as Figure 4.4 demonstrates, is very close to the local ones. Therefore, exploring the search space properly is vital in such a problem. Other classic test problems are the *Griewank* (f_4) and *Rastrigin* (f_5) functions. Both, as Figure 4.5 and Figure 4.6 exemplify, have various local minima and only one global solution. Unlike the *Schaffer* function, the dimensions of the remaining test functions are relatively high, which in turn result in further complexity.

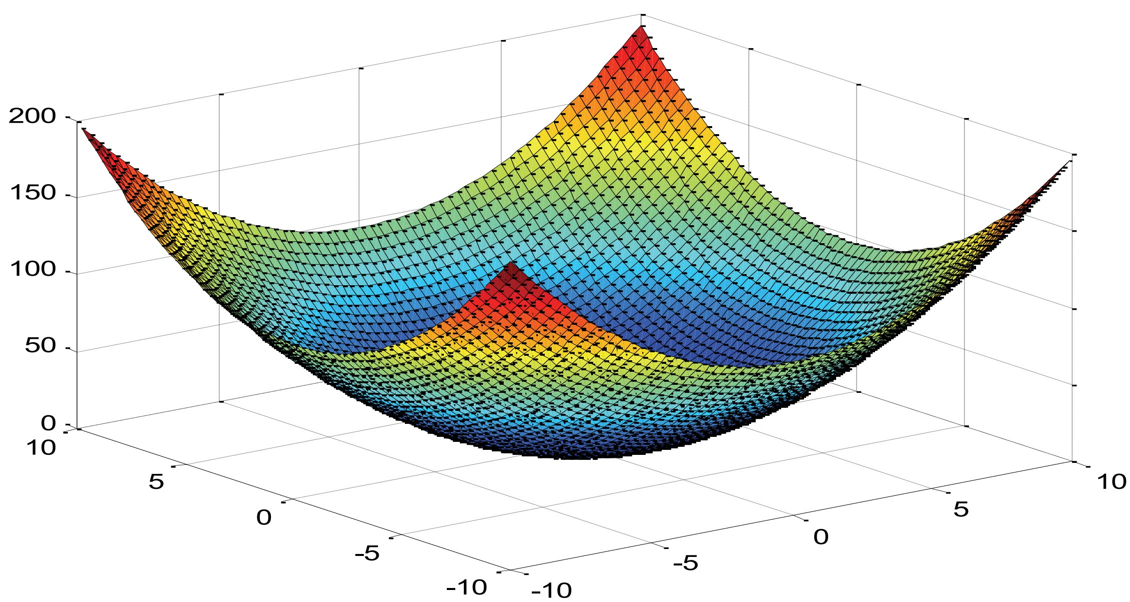


Figure 4. 2: 3D plot of the *Sphere* (f_1) function.

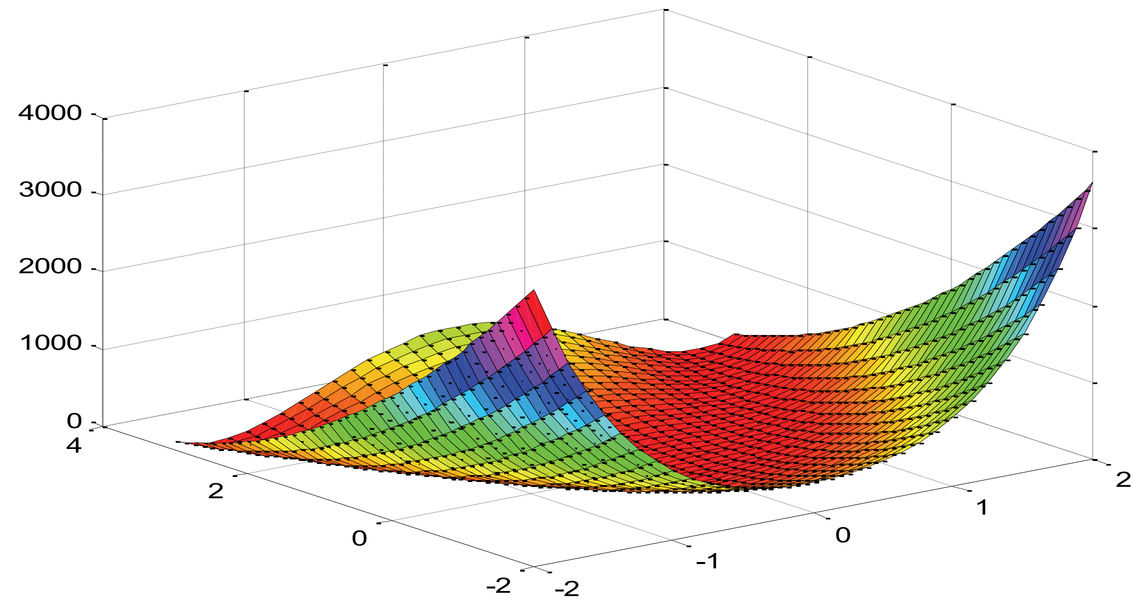


Figure 4. 3: 3D plot of the *Rosenbrock* (f_2) function.

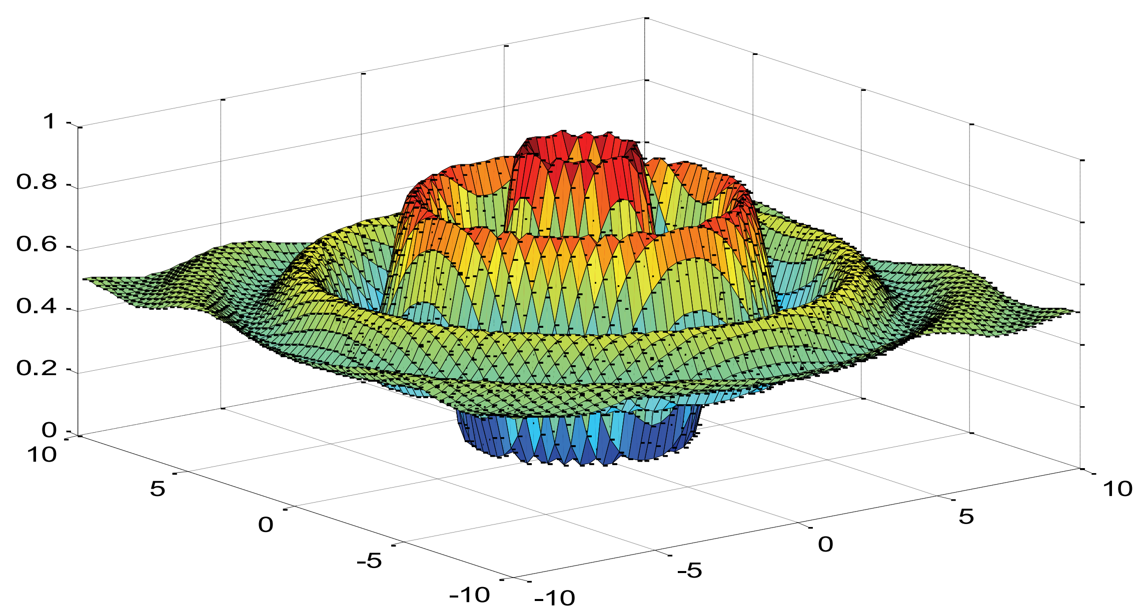


Figure 4. 4: 3D plot of the *Schaffer* (f_3) function.

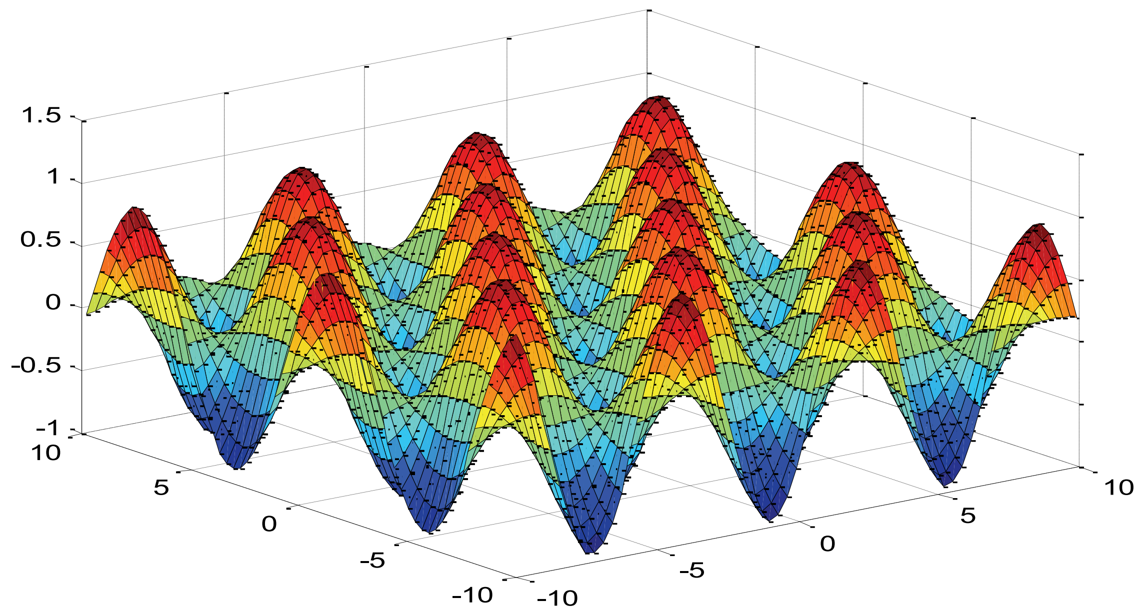


Figure 4. 5: 3D plot of the *Griewank* (f_4) function.

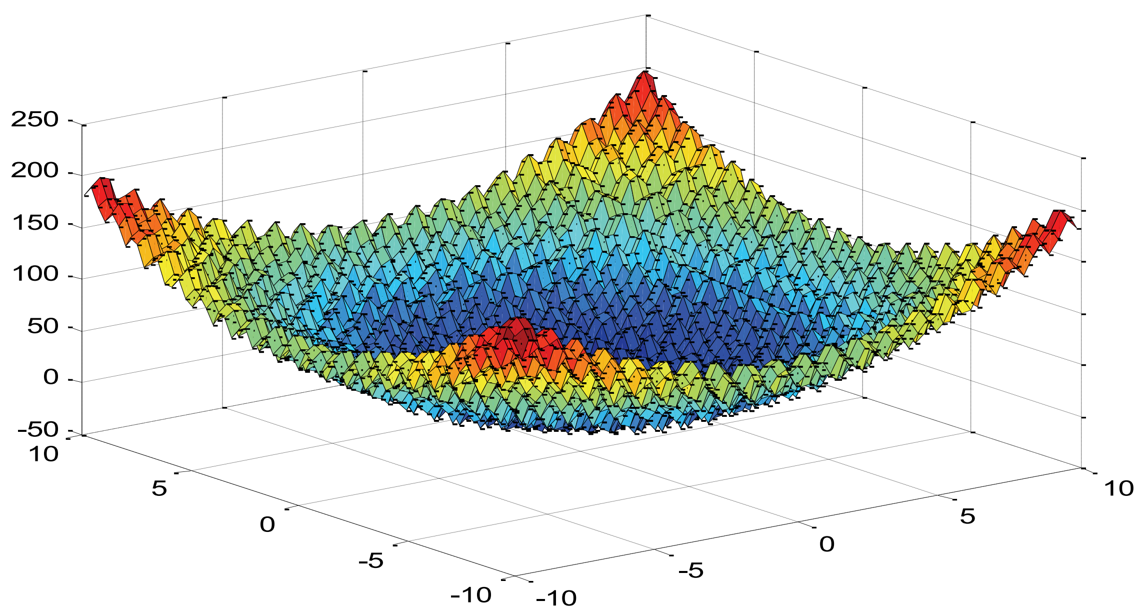


Figure 4. 6: 3D plot of the *Rastrigin* (f_5) function.

The global minima for all selected functions are equal to zero and the optimal values of (x_1^*, …, x_D^*) are zeros as well, except for the (f_2) function where the values of (x_1^*, …, x_D^*) are ones. Clearly, as indicated in Table 4.2, the results of the proposed

SDOA outperformed those obtained using other algorithms [89]. In addition, the required CPU time of the proposed algorithm to obtain a solution is considered fast.

Table 4. 2: Statistical measurements of the selected benchmark functions; Results of the proposed algorithm are emphasized in boldface; Std.Dev.: Standard Deviation

f_i	Statistic	GA	PSO	DE	ABC	**SDOA**	**CPU (s)**
f_1	Mean	1.11×10^{3}	0.0000	0.0000	0.0000	**0.0000**	**0.4**
	Std.Dev.	74.2145	0.0000	0.0000	0.0000	**0.0000**	
f_2	Mean	1.96×10^{5}	15.0886	18.204	0.0888	**0.0170**	**0.4**
	Std.Dev.	3.85×10^{4}	24.1702	5.0362	0.0774	**0.0167**	
f_3	Mean	0.00424	0.0000	0.0000	0.0000	**0.0000**	**0.2**
	Std.Dev.	0.00476	0.0000	0.0000	0.0000	**0.0000**	
f_4	Mean	10.6335	0.01739	0.0015	0.0000	**0.0000**	**1.1**
	Std.Dev.	1.16146	0.02081	0.003	0.0000	**0.0000**	
f_5	Mean	52.9226	43.9771	11.7167	0.0000	**0.0000**	**0.9**
	Std.Dev.	4.56486	11.7287	2.5382	0.0000	**0.0000**	

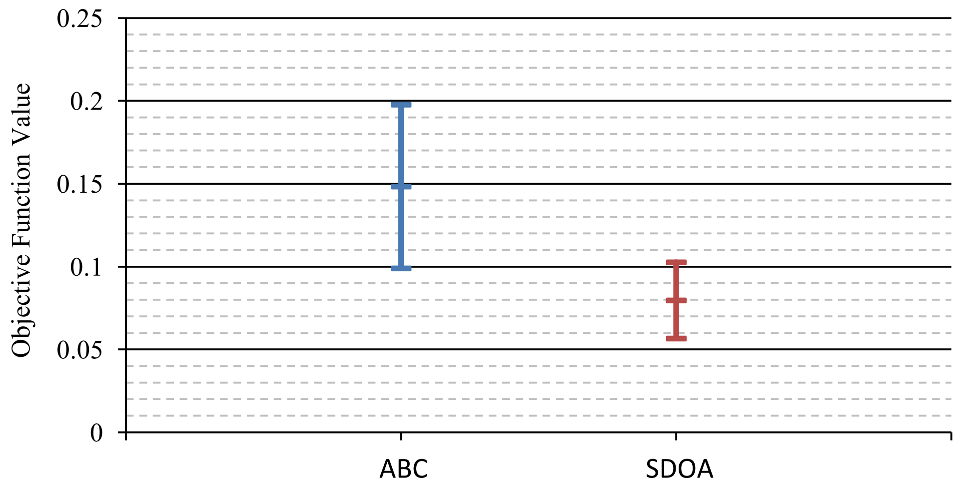
The classical *Rosenbrock* (f_2) function, among those selected test functions, is considered a challenge. Indeed, this function has a flat parabolic valley with non-convex characteristics. Although finding the valley is trivial, approaching the global solution inside this narrow surface is known to be difficult [87]. Accordingly, and in order to confirm the efficiency of the proposed algorithm performance, the *Rosenbrock* (f_2) function was designated for further simulation by increasing the dimension to 100. The population size remained unchanged but the maximum number of evaluations was constrained at 1×10^{6}.

The statistical measurements, as seen in Table 4.3 and Figure 4.7, proved that the solution's quality of the SDOA outperformed those of ABC method. However, the required CPU times of the SDOA were slightly higher than the ABC algorithm.

Table 4. 3: Statistical measurements of the proposed SDOA applied to the *Rosenbrock* function (f_2) with 100 dimensions; Min: Minimum; Max: Maximum; Std.Dev.: Standard Deviation.

x-Range	Method	Min.	Max.	Mean	Std.Dev.	CPU (s)
[-30,30]	ABC	0.01022	0.28637	0.14830	0.13807	1.1
	SDOA	0.01210	0.16596	0.07956	0.06423	1.3
[-100,100]	ABC	0.47869	0.97742	0.72805	0.24936	1.7
	SDOA	0.02190	0.36390	0.16801	0.14398	2.1

95% Confidence Interval with [-30, 30] *x*-Range



95% Confidence Interval with [-100, 100] *x*-Range

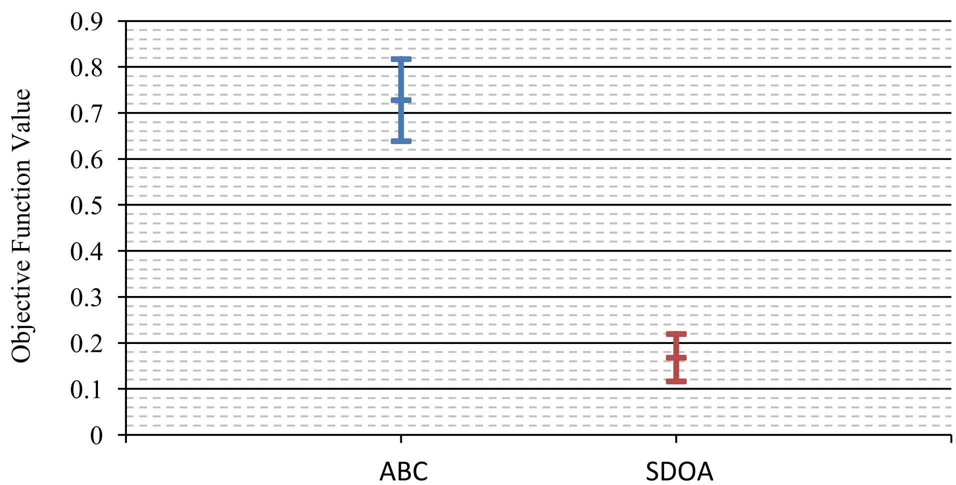


Figure 4. 7: The 95% confidence intervals of ABC and SDOA methods with respect to different search-space ranges.

4.4 SUMMARY

In this chapter, a new population-based sensory-deprived optimization algorithm (SDOA) was proposed to solve numerical optimization problems. Its distinct and main advantage occurred in the semi-exploitation and semi-exploration processes. Thus, the SDOA covered a wide range of the solution search-space, and evaded premature convergence. The solution vector from the temporary semi-exploration routine enhanced the solution vector of the main algorithm.

A set of benchmark optimization functions was used to evaluate the performance of the proposed SDOA technique and compare its results with those attained using other well-known algorithms. In addition, the SDOA was utilized in a mixed integer nonlinear optimization applications in [157]. The efficiency of the SDOA method was confirmed by the fact that the standard deviation of the results attained for 30 independent runs was virtually zero. The performance of the presented SDOA algorithm addressed its superiority and the potential for solving complex and larger dimensional problems in future research.

CHAPTER 5: COMPUTATIONAL RESULTS

5.1 INTRODUCTION

The majority of existing prime movers in electrical power systems are fossil fuel based. The ultimate goal is to meet the required load demand at the lowest operating costs subject to satisfying various practical equality and inequality constraints. Environmental concerns, due to Green House Gas (GHG) emissions caused by fossil fuels, affect the achievement of this goal and power utilities strive to minimize costs and emissions simultaneously.

Power utilities control or dispatch centres prepare (solve) the next-day dispatch schedule for the predetermined-committed units, after utilizing the day-ahead forecasted load demand. The solution of the DED is based on the assumption that the load demand is a constant between sampling points, but it is not. Therefore, utilities overcome this drawback by performing (updating) the DED solution from every few minutes to one hour. Although available spinning reserve plays a major role in utilities' operation in terms of stability and reliability, it will increase operational costs. Most utilities bypass or "limit" this additional operating cost by contracting one or more consumers (usually large ones) to be off the utility's generation supply. This approach has more customer satisfaction than the identical load shedding; nevertheless, load shedding practices exist as a last resort.

In this chapter, the ABC algorithm is utilized to solve the DED and DEED problems. In addition, a new constrained search-tactic is offered to enhance the algorithm's performance. The SDOA, in this chapter, is adopted to solve two test systems for two reasons: 1) to validate the efficacy of the proposed constrained search-tactic with another algorithm, and 2) to show the potential of the SDOA in solving such a complex, non-convex, and nonlinear optimization problem. Therefore, the performance of the SDOA and ABC algorithms with and without the integration of the constrained search-tactic are evaluated by solving different test cases. The test systems are categorized into two groups according to the objective functions considered. The first one is used to solve single objective function problems as well as to confirm the proposed constrained search-tactic, while the second group represents the multiple objective functions problems. Analyzing the effect of integrating renewable energy sources in the multiple objective functions problem is also considered in the second group. This chapter is organized as follows: Section 5.2 explains the mathematical formulation of the optimization problems considered; Section 5.3 clarifies the proposed constrained search-tactic; Section 5.4 demonstrates the computational results and discussion of the test systems; Section 5.5 highlights the conclusions.

5.2 MATHEMATICAL FORMULATION

Since the objective function of the DED problem is to minimize the operating fuel's costs of committed generating units to meet the load demand, subject to equality and inequality constraints over a predetermined dispatch period, the result's practical usefulness will be degraded if the units' valve-point effects are neglected. Consequently, there are two models to represent the units' valve-point effects in the literature [48]. The first represents the units' valve-point effects in terms of prohibited operating zones which are included as inequality constraints. The second form represents the units' valve-point effects as a rectified sinusoid term which is superimposed on the approximate quadratic fuel cost function. Figure 5.1 demonstrates the cost function of the generator unit based on different forms as well as the classical quadratic form. The general mathematical form of the DED problem is as follows:

$$\boldsymbol{F} = min \sum_{t=1}^{T} \sum_{i=1}^{N} f_i \left(P_i^t\right) \tag{5.1}$$

where, f_i (P_i) is the fuel cost function of i^{th} generator and P_i^t is the output power of i^{th} generator at a time t, $\forall$ $i \in \{1, 2, \ldots, \mathrm{N}\}$ and $\forall$ $t \in \{1, 2, \ldots, \mathrm{T}\}$. Equations (5.1.1) and (5.1.2) [158] express the fuel cost function neglecting and considering the units' valve-point effects, respectively.

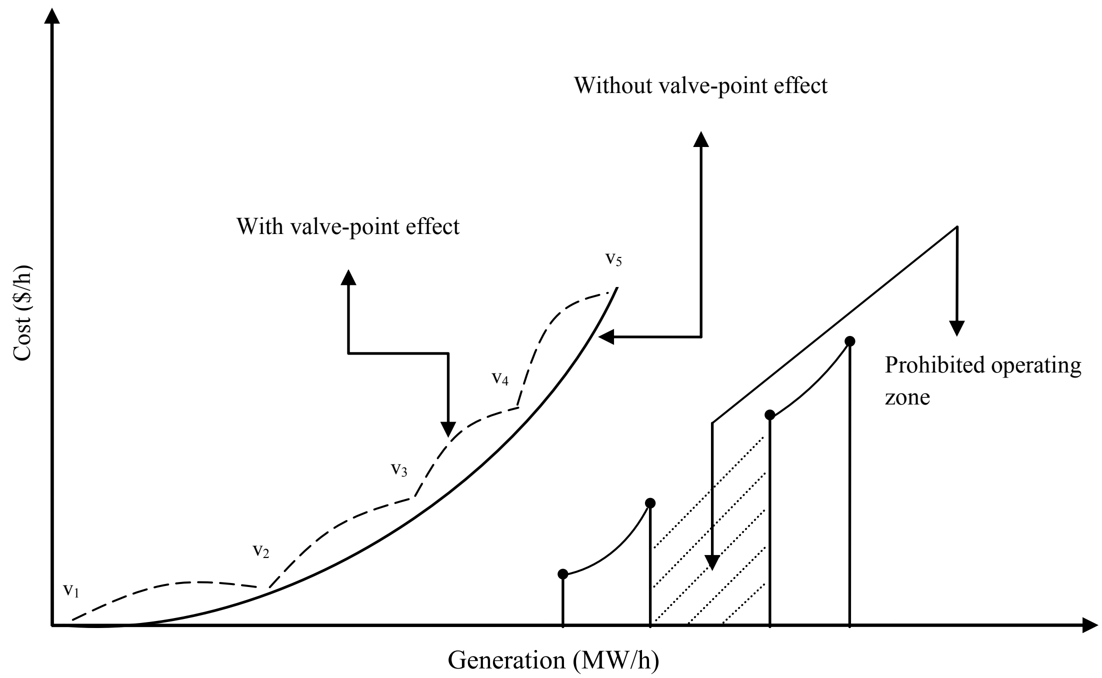


Figure 5. 1: Output power response of generator unit due to different fuel cost functions; dashed and solid curves represent non-smooth and smooth fuel cost functions, respectively.

$$f_i(P_i) = a_i + b_iP_i + c_iP_i^2 \tag{5.1.1}$$

$$f_i(P_i) = a_i + b_iP_i + c_iP_i^2 + \left|d_i \times sin\left(g_i \times \left(P_{i,min} - P_i\right)\right)\right| \tag{5.1.2}$$

where, a_i, b_i, c_i, d_i, and g_i are the i^{th} generator's coefficients, and $P_{i\ min}$ is the minimum limit of i^{th} generator.

Due to environmental consciousness, the classical goal of power utility, i.e., dispatching with the least fuel costs, is hampered. Power plants using fossil fuels have to manage the gases' emissions, e.g., carbon dioxide (CO_2), nitrogen oxides (NO_x), and sulfur dioxide (SO_2). Different solutions have been proposed in [159] to deal with the emissions of power plants:

1. Integrate filters to clean harmful gases attributable to the combustion process.
2. Operate with (cleaner) fuels that contain lower emissions.
3. Superimpose the emissions' impact at the optimal dispatch process.

Considering the third option from a cost-effective standpoint, emissions pollutants have been incorporated in the optimization problem either as competing with the objective function [31, 33, 51], or as an inequality constraint [15, 34, 52]. In this thesis, the former approach is adopted to represent the DEED problem. Therefore, the quantity of emissions pollutants per generation units can be expressed as follows [9]:

$$\boldsymbol{E} = min \sum_{t=1}^{T} \sum_{i=1}^{N} E_i \left(P_i^t\right) \tag{5.2}$$

$$E_i(P_i) = \alpha_i + \beta_i P_i + \gamma_i {P_i}^2 + \eta_i \, e^{\delta_i P_i} \tag{5.2.1}$$

where, E_i (P_i) is the emissions function of i^{th} generator, α_i, β_i, γ_i, η_i, and δ_i are the i^{th} generator's coefficients, e is an exponential term, and P_i is the output real power of i^{th} generator.

Moreover, it is impractical to neglect the transmission losses of the system, so the *B*-coefficient formula is commonly used to express it. Thus, the real power balance equation representing equality constraints of the problem considered is as follows:

$$\sum_{t=1}^{T} \sum_{i=1}^{N} P_i^t = \sum_{t=1}^{T} P_D^t + P_L^t \tag{5.3}$$

where, P_D^t and P_L^t are the load demand and system's loss at a time t respectively. Integration of a renewable source (*RS*) modifies the equality constraints function [160] to be as follows:

$$\sum_{t=1}^{T} \sum_{i=1}^{N} P_i^t = \sum_{t=1}^{T} \left(P_D^t + P_L^t - \sum_{RS=1}^{M} \mu_{RS}^t \, P_{RS}^t \right) \tag{5.3.1}$$

$$\sum_{t=1}^{T} \sum_{i=1}^{N} \left| P_i^t - P_D^t - P_L^t \right| \le \sum_{t=1}^{T} \sum_{RS=1}^{M} \left| \mu_{RS}^t \, P_{RS}^t \right| \tag{5.3.2}$$

where, μ_{RS}^t is a multiplier set to a permissible amount of active power injected by *RS* at time t, P_{RS}^t is the forecasted real power from *RS* at time t $\forall$ $RS \in \{1, 2, \ldots, M\}$. In this thesis, μ_{RS} is set to one, and $t \in \{1, 2, \ldots, T\}$.

The system's active power loss can be calculated using Kron's loss formula [161] as follows:

$$\sum_{t=1}^{T} P_L^t = B_{00} + \sum_{t=1}^{T}\sum_{i=1}^{N} B_{i0}P_i^t + \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{j=1}^{N} P_i^t B_{ij} P_j^t \tag{5.4}$$

Alternatively, by omitting the first two terms of (5.4), the system's active power loss can be calculated using George's loss expression [161] as follows:

$$\sum_{t=1}^{T} P_L^t = \sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{j=1}^{N} P_i^t B_{ij} P_j^t \tag{5.5}$$

where, B_{00} is the loss coefficient constant; B_{i0} is the i^{th} element of the loss coefficient vector; B_{ij} is the ij^{th} element of the loss coefficient square matrix. It is important to mention that (5.4) and (5.5) are approximate formulas representing the system's active power loss. In addition, these two equations are utilized under the assumption that intense changes to the system's status have not occurred [162, 163]. Although power flow calculation offers system's power loss in a more detailed model, it creates an extra burdensome to the considered problem. Therefore, Kron's and George's loss's expressions are favoured [161].

The inequality constraints of the DED and DEED problems are the units' ramp-rate limits, i.e., upper rate (UR_i) and down rate (DR_i), are considered as follows:

$$\begin{aligned} P_i^t - P_i^{t-1} &\le UR_i \\ P_i^{t-1} - P_i^t &\le DR_i \end{aligned} \tag{5.6}$$

Additional inequality constraints are the minimum and maximum power output of each unit:

$$P_{i,min} \le P_i \le P_{i,max} \tag{5.7}$$

Therefore, to incorporate the constraints of units' ramp-rate limits (5.6) in the real power output limit constraints (5.7), the modified units' real power outputs are evaluated [49] as follows:

$$
\begin{aligned}
P_{i,min}^{t} &= max\left(P_{i,min}, P_i^{t-1} - DR_i\right) \\
P_{i,max}^{t} &= min\left(P_{i,max}, P_i^{t-1} + UR_i\right)
\end{aligned} \tag{5.8}
$$

The following inequality constraints describe the case when units have prohibited operating zones [30] defined by:

$$
\begin{cases}
P_{i,min} \le P_i^t \le P_{i,1}^l \\
P_{i,j-1}^u \le P_i^t \le P_{i,j}^l \\
P_{i,n_i}^u \le P_i^t \le P_{i,max} & ,j \in \{2, 3, \ldots, n_i\}
\end{cases} \tag{5.9}
$$

where, $P^{l}_{i,1}$ is the lower limit of the first prohibited zone of the i^{th} generator; $P^{u}_{i,j-1}$ is the upper limit of the $(j\text{-}1)^{th}$ prohibited zone of the i^{th} generator; $P^{u}_{i,ni}$ is the upper limit of the n^{th} prohibited zone of the i^{th} generator; n_i is the number of prohibited zones in the i^{th} generator.

5.3 NEW CONSTRAINED SEARCH-TACTIC

The challenging level of optimization problems increases when a large number of equality and inequality constraints must be satisfied. One way to handle such a problem is by transferring it from a constrained formed to an unconstrained one. The penalty factor is a frequently used method to include those constraints into the objective function. However, each constraint has a characteristic degree of dominance affecting the algorithm performance, and – somehow – directs the algorithm towards the optimal or quasi-optimal region. The proposed search-tactic utilizes those constraints to accelerate the algorithm performance towards the optimal feasible region.

The following procedures describe the proposed constrained search-tactic for the scenario of a one-hour dispatch period. Only steps 2 and 3 are utilized in the one-day-ahead dispatch period scenario, i.e., when multiple time intervals are considered.

1. The maximum and minimum output powers of each unit are modified to incorporate the constraints of the units' ramp-rate limits [49], and updated in every dispatch (t) hour using (5.8).
2. The objective function is altered "temporarily" from minimizing the operating fuel costs and/or emissions to minimizing the violation of the real power

balance equation. This substitution of objective function is applied at the initial population calculations. Once the (φ + 1) loop (cycle) starts, the main objective function is retained until the stopping role is met. The idea behind this substitution is to avoid exposing too many infeasible solutions. The value of (φ) is chosen to be 10% of the maximum iteration number of an algorithm.

3. The handling mechanism for the units' prohibited operating zones is as follows. Consider a unit (*i*) in a solution vector (x_i) operates at a time (*t*) within a prohibited operating zone (*j*); the following procedures describe the handling method for the units' prohibited operating zones:

3.1 Divide the solution (x_i) into two sub-solutions by only modifying the unit (i) output according to its violated prohibited operating zone (j).

3.2 Force each one of the two sub-solutions to adjust unit (i) output to operate in its permissible upper and lower limits of the associated prohibited operating zone (j), while taking into account the unit's ramp-rate constraints.

3.3 Evaluate both sub-solutions, and select the best based on a greedy-selection method.

For the case when multiple units (*y*) violate their prohibited operating zones' constraints, the number of sub-solutions equals 2^y. The main advantage of the proposed search-tactic is reducing (significantly) the degree of randomness in the initial population – consequently, accelerating the algorithm's performance towards the optimum feasible region.

5.4 SIMULATIONS AND RESULTS

Various test systems are examined to verify the validity of the algorithms offered. The acceptable violation of equality constraints (power balance equation) is adjusted to be less than or equal 10^{-4}. The control parameters of selected algorithms are tuned after trial-and-error experiments. The proposed methods are implemented in C language, and executed on an Intel® core™ 2 duo PC with 2.66-GHz speed and 4GB RAM. The results obtained after carrying out 30 independent runs are compared with those attained using other well-known techniques. The solutions' quality and required CPU time demonstrated the efficiency of the offered methods.

5.4.1 Single Objective Function Problem

In this set of test systems, the objective function is to minimize the operating fuel costs (or emissions) of committed generation units to meet the load demand, subject to various equality and inequality constraints. Three systems frequently used in the literature are adopted to evaluate the performance of the proposed ABC algorithm. In addition, the effectiveness of the offered constrained search-tactic is validated in this set of test systems as well as via utilizing the SDOA algorithm by solving two of those systems. A comparison between the results attained by the suggested methods, and those obtained using different techniques is conducted. The first two cases involve the scenario where the DED problem is designated for a one hour dispatch schedule. A one-day dispatch schedule is represented in the remaining cases where the ABC algorithm is utilized.

5.4.1.1 Case 1-A

Table A.1 lists the system's six generating units' characteristics and coefficients. The objective function was to operate these units economically to meet the 1.263 GW load demand. The considered constraints in this system were the transmission losses, units' ramp-rate limits, units' bounded limits, as well as double prohibited operating zones in each unit. The *B*-loss coefficient matrices reported in [40] and listed in Table A.2. The ABC and SDOA algorithms were adopted to solve this system.

The ABC parameters for this system: *CS*, *limit*, and *MCN* were tuned as 100, 300, and 300 respectively. The SDOA parameters for this system: *SDP*, $SDP_{parallel}$, iteration number, and *R* were adjusted as 30, 3, 300, and 36 correspondingly. Due to integrating the proposed constrained search-tactic, the objective function has been altered for the first 10% iteration for both algorithms, i.e., the φ parameter adjusted as 30 per algorithm.

As shown in Table 5.1, the attained results of ABC and SDOA algorithms (with and without the integration of the proposed search-tactic) presented no violations to the problem's constraints while meeting the load demand sufficiently. The proposed search-tactic enhanced the performances of both algorithms. In other words, the integration of the presented search-tactic obtained an average 43% reduction in the required CPU time. Although the total fuel costs listed in Table 5.1 were virtually identical, the integration of the suggested tactic in both algorithms successfully attained lower operating costs. The

SDOA* (with the integration of the offered search-tactic) represented the lowest operating cost for this system. On the other hand, the ABC* (with the integration of the offered search-tactic) resulted in outperforming the SDOA* method in terms of less CPU time. The constrained search-tactic, as shown in Figure 5.2 and Figure 5.3, prevented both algorithms from exposing infeasible solutions at initial iterations. It, consequently, accelerated both algorithms' performances toward the optimal region.

Table 5. 1: Optimal dispatch power for case 1-A; Sys.Viol.: System's Power Balance Constraints' Violation.

Unit	Techniques			
	ABC	ABC*	SDOA	SDOA*
P_1 (MW)	448.12800	448.12600	447.37700	447.96000
P_2 (MW)	172.60900	173.49900	172.78900	173.67200
P_3 (MW)	262.58600	262.60000	262.65500	263.55900
P_4 (MW)	137.45900	136.96900	138.25000	139.46200
P_5 (MW)	168.19700	167.54900	167.59700	164.73500
P_6 (MW)	87.03400	87.270000	87.32400	86.55800
Total output power (MW)	1,276.01300	1,276.01300	1,275.99200	1,275.94600
Total P_L (MW)	13.01300	13.01300	12.99200	12.94600
Total system's power loss (%)	1.020	1.020	1.018	1.015
|Sys.Viol.| (MW)	0.00000	0.00000	0.00000	0.00000
Total operating costs ($/h)	15,450.01	15,449.99	15,449.96	<u>15,449.91</u>
Average CPU time (s)	0.552	<u>0.319</u>	0.843	0.478

* With the integration of the proposed constrained search-tactic.

For comparison purposes, the results of the proposed search-tactic via ABC and SDOA algorithms were compared with those obtained using other well-known algorithms, such as SA, GA, TS, PSO, MTS, DE, HS, hybrid harmony search (HHS), and evolution strategy optimization (ESO) reported in [40] and [164]. The statistical measurements, as shown in Table 5.2, were attained after conducting 30 independent runs with different initials for the six-unit system. It is important to reveal that the system's power balance constraints' violations reported in [40] and [164] is considered large except for the solutions obtained by the GA and PSO algorithms. On the other hand, the results obtained via utilizing the proposed search-tactic in ABC and SDOA algorithms avoided such local and infeasible solutions. The large power mismatch of equality constraints reported in Table 5.2 degraded the least cost solution attained by the ESO method. Although the solutions obtained using DE and HS methods presented more robustness in the convergence characteristics, they in addition to HHS method suffered from the infeasibility of the outcomes obtained. In contrast, the ABC* and SDOA* algorithms successfully obtained the economic operating output power dispatch for this system without violating the system's equality and inequality constraints. The proposed algorithms outperformed all the solutions reported in [40] in terms of minimum fuel cost, solution quality, and required CPU time. An average 0.10% reduction in the operating fuel's cost was obtained by the ABC* and SDOA*, and significant reductions in the required CPU time were attained by the ABC* (95%) and SDOA* (93%) tactics with respect to the PSO method. Although the reduction in operating fuel's cost was small, its significance would lead to considerable annual cost saving.

Table 5. 2: Comparison of results of the proposed tactic for case 1-A; Max: Maximum; Avg.: Average; Min: Minimum; Sys.Viol.: System's Power Balance Constraints' Violation; Std.Dev.: Standard Deviation; --: Not Available.

Method	Max. ($/h)	Avg. ($/h)	Min. ($/h)	|Sys.Viol.| (MW)	Std.Dev.	CPU (s)
SA[a]	15,545.50	15,488.98	15,461.10	0.00190[a]	28.3678	50.360[a]
GA[a]	15,524.69	15,477.71	15,457.96	0.00010[a]	17.4072	46.600[a]
TS[a]	15,498.05	15,472.56	15,454.89	0.00620[a]	13.7195	20.550[a]
PSO[a]	15,491.71	15,465.83	15,450.14	0.00000[a]	10.1502	6.820[a]
MTS[a]	15,453.64	15,451.17	15,450.06	0.00260[a]	0.92870	1.290[a]
DE[b]	15,450.00	15,450.00	15,450.00	0.02000[b]	0.00000	0.033[b]
HS[b]	15,449.00	14,449.00	15,449.00	0.07200[b]	0.00000	6.830[b]
HHS[b]	15,453.00	15,450.00	15,449.00	0.04000[b]	--	0.140[b]
ESO[b]	15,470.00	15,430.00	15,408.00	3.37000[b]	--	0.360[b]
ABC*	15,451.80	15,451.19	15,449.99	0.00000	0.66110	0.319
SDOA*	15,452.64	15,451.30	15,449.91	0.00000	0.88880	0.478

* With the integration of the proposed constrained search-tactic; [a] As reported in [40]; [b] As reported in [164].

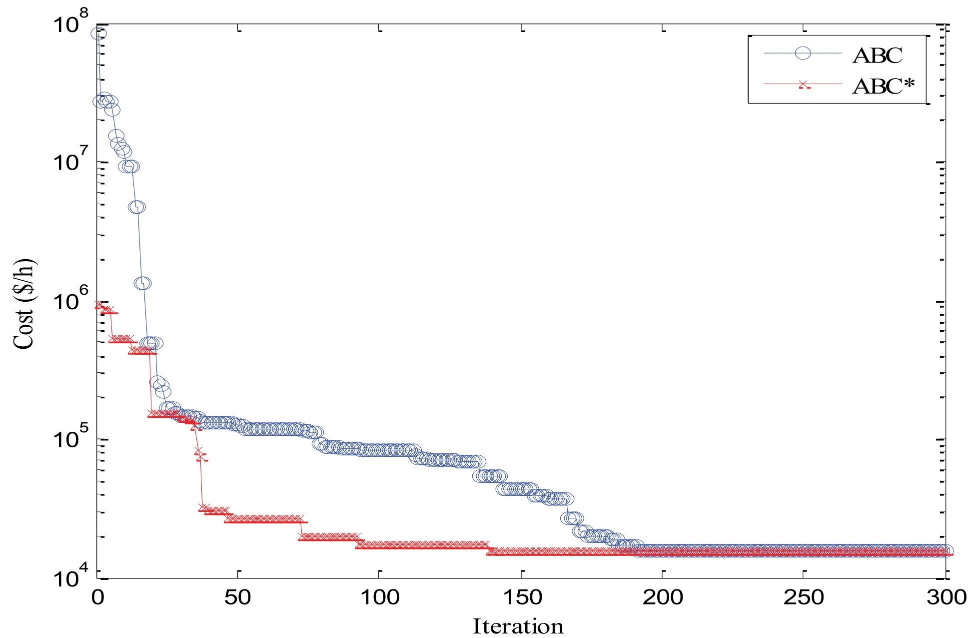


Figure 5. 2: The ABC algorithm's performance for case 1-A.

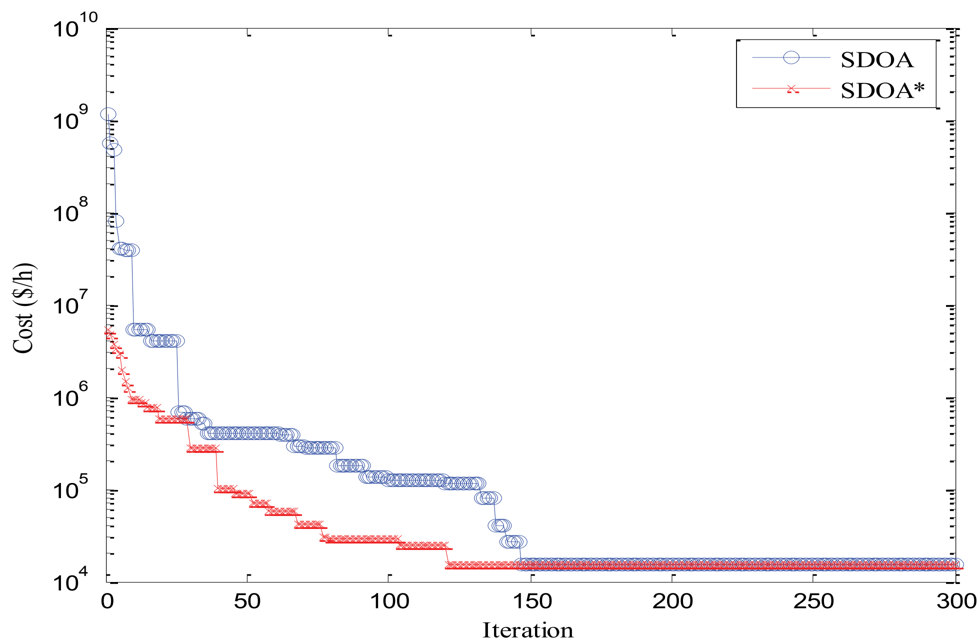


Figure 5. 3: The SDOA algorithm's performance for case 1-A.

5.4.1.2 CASE 2-A

This system includes 15 generating units with the data being recorded in Table A.3. The constraints considered were the units' prohibited operating zones, units' ram-rate limits, transmission losses, and units' output power limits. Data of the *B*-loss coefficient matrices reported in [40] as well as in Table A.4. The load demand for this system was a 2.630 GW. Once more, the ABC and SDOA algorithms were designated to solve this system.

The ABC parameters for this system: *CS*, *limit*, and *MCN* were tuned as 200, 15×10^2, and 500 respectively. The SDOA parameters for this system: *SDP*, $SDP_{parallel}$, iteration no., and *R* were adjusted as 50, 5, 500, and 225 respectively. The φ parameter was tuned as 50 in both algorithms.

As recorded in Table 5.3, utilizing the proposed search-tactic improved both algorithms' performances. An average of 31% reduction in the CPU required time was due to the integration of the offered tactic. The constrained search-tactic accelerated both algorithms' performances as in Figure 5.4 and Figure 5.5. Among all the results in Table 5.3, the SDOA* attained the most economic output power dispatch for the 15-unit system. In addition, the SDOA* outperformed the ABC* method with respect to the required CPU time.

The results obtained using the proposed algorithms were compared with those of other algorithms reported in [40]. Statistical measurements obtained after carrying out 30 independent runs with different seeds, shown in Table 5.4, clarified that the results attained using the search-tactic offered, outperformed those of other well-known algorithms. Even though the GA method did not demonstrate a violation to the system equality constraints, it is trapped in a local minimum solution. The outcomes of the ABC* and SDOA* algorithms attained better results with regard to fuel's cost, solution quality, and required CPU time. An approximately 98% reduction in the required CPU time was achieved by the ABC* and SDOA* algorithms compared to the GA method. In addition, an average 0.40% reduction in the operating fuel's cost obtained by the ABC* and SDOA* algorithms with respect to the GA method. Although the reduction in operating fuel's cost was smaller than that of CPU time, its significance would lead to considerable annual cost saving. It is important to highlight that the obtained results of both proposed algorithms successfully satisfied the problem's equality and inequality constraints.

Table 5. 3: Optimal dispatch power for case 2-A; Sys.Viol.: System's Power Balance Constraints' Violation.

Unit	Techniques			
	ABC	ABC*	SDOA	SDOA*
P_1 (MW)	454.99500	455.00000	455.00000	455.00000
P_2 (MW)	380.00000	380.00000	380.00000	380.00000
P_3 (MW)	130.00000	130.00000	130.00000	130.00000
P_4 (MW)	130.00000	130.00000	130.00000	130.00000
P_5 (MW)	169.97300	170.00000	170.00000	170.00000
P_6 (MW)	460.00000	460.00000	460.00000	460.00000
P_7 (MW)	430.00000	430.00000	430.00000	430.00000
P_8 (MW)	71.97000	71.97000	71.97000	71.66300
P_9 (MW)	59.18000	59.18000	59.18000	59.18000
P_{10} (MW)	159.53000	159.49600	159.80000	159.80000
P_{11} (MW)	80.00000	80.00000	79.71900	80.00000
P_{12} (MW)	80.00000	80.00000	79.99800	80.00000
P_{13} (MW)	25.00200	25.00200	25.00200	25.00200
P_{14} (MW)	15.00600	15.00600	15.00600	15.01200
P_{15} (MW)	15.00100	15.00100	15.00100	15.00100
Total output power (MW)	2,660.65700	2,660.65500	2,660.67600	2,660.65800
Total P_L (MW)	30.65700	30.65500	30.67600	30.65800
Total system's power loss (%)	1.152	1.152	1.153	1.152
|Sys.Viol.| (MW)	0.00000	0.00000	0.00000	0.00000
Total operating costs ($/h)	32,704.52	32,704.48	32,704.81	<u>32,704.47</u>
Average CPU time (s)	1.121	0.788	1.057	<u>0.709</u>

* With the integration of the proposed constrained search-tactic.

Table 5. 4: Comparison of results of the proposed tactic for case 2-A; Max: Maximum; Avg.: Average; Min: Minimum; Sys.Viol.: System's Power Balance Constraints' Violation; Std.Dev.: Standard Deviation.

Method	Max. ($/h)	Avg. ($/h)	Min. ($/h)	|Sys.Viol.| (MW)	Std.Dev.	CPU (s)
SA[a]	33,028.95	32,869.51	32,786.40	0.01170[a]	112.32	71.250[a]
GA[a]	33,041.64	32,841.21	32,779.81	0.00000[a]	81.220	48.170[a]
TS[a]	32,942.71	32,822.84	32,762.12	0.12800[a]	60.590	26.410[a]
PSO[a]	32,841.38	32,807.45	32,724.17	0.02990[a]	21.240	13.250[a]
MTS[a]	32,796.15	32,767.21	32,716.87	0.01120[a]	17.510	3.650[a]
ABC*	32,995.87	32,723.78	32,704.48	0.00000	51.382	0.788
SDOA*	32,959.27	32,723.47	32,704.47	0.00000	45.771	0.709

* With the integration of the proposed constrained search-tactic; [a] As reported in [40].

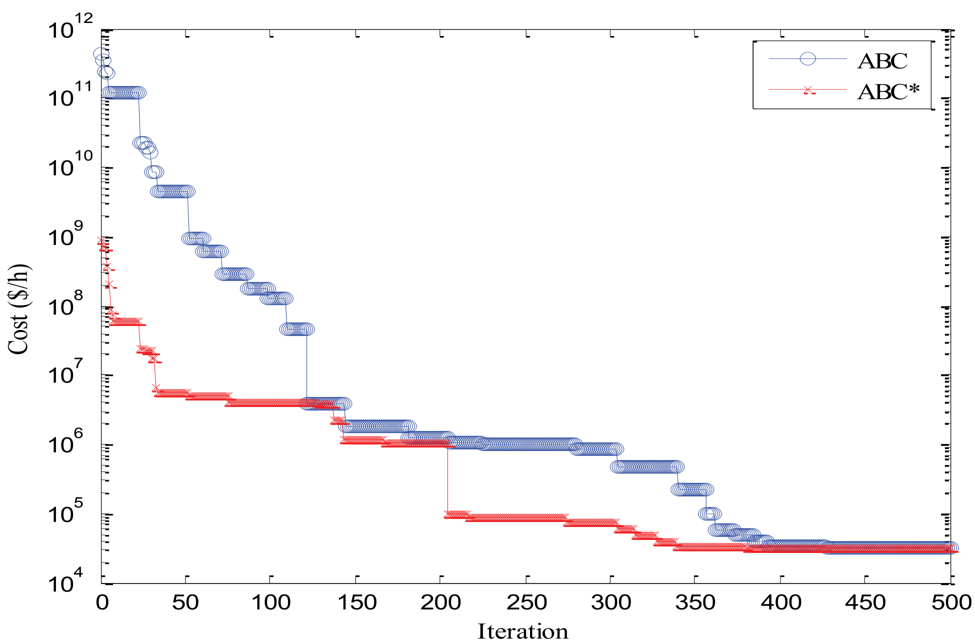


Figure 5. 4: The ABC algorithm's performance for case 2-A.

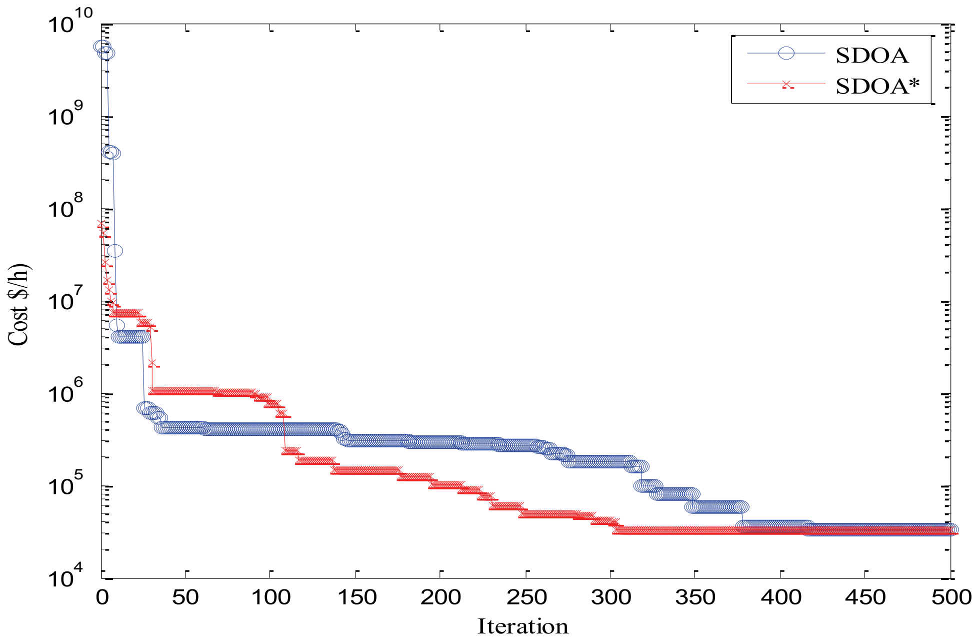


Figure 5. 5: The SDOA algorithm's performance for case 2-A.

5.4.1.3 CASE 3-A

As recorded in Table A.5, the aim of this system was to operate its five generating units economically to meet the 24-hour load demand subject to satisfying various equality and inequality constraints. This system's constraints were the transmission losses using George's loss expression, units' bound limits, and unit's ramp-rate limits. The operating fuel cost's function superimposed the unit's prohibited operating zones by a rectified sinusoid term. The system's load demand and *B*-loss coefficient matrix were reported in Table A.6 and Table A.7 respectively. The ABC algorithm designated to solve this system with and without the integration of the proposed constrained search-tactic.

After trail-and-error experiments, the ABC parameters for this system increased dramatically due to the increase in the problem's dimension. Therefore, the ABC's *CS*, *limit*, and *MCN* were tuned as 300, 18×10^3, and 30×10^3 respectively. Integrating the proposed constrained search-tactic will alter the main objective function for the first 3×10^3 (φ) iterations.

As presented in Table 5.5, the integration of the proposed constrained search-tactic enhanced the ABC algorithm's performance resulting in a significant reduction (4.41%) in the operating fuel's cost. The cost saving ($2,254.60 daily) is shown in Figure 5.6. An average 21% reduction in the required CPU time was obtained as well. The load demand curve, each committed unit's output power, and the total system's input power are exemplified in Figure 5.7. The optimal dispatched power for this system guaranteed that the system's constraints were satisfied during the 24-hour dispatch period, as shown in Table 5.6 and Figure 5.7. In other words, the output power of each unit in every time interval was consistent with the output power of adjacent units. The ABC algorithm's performance for this system is in Figure 5.8.

Table 5. 5: Comparison of results of the proposed ABC algorithm for case 3-A; Max: Maximum; Avg.: Average; Min: Minimum; Corresp.: Corresponding emission to the min ($); Std.Dev.: Standard Deviation.

Method	Max. ($)	Avg. ($)	Min. ($)	Corresp. (lb)	Std.Dev.	CPU (s)
ABC	51,868.90	51,462.82	51,102.80	20,407.8	229.191	280.440
ABC*	50,195.90	49,814.28	**48,848.20**	23,413.9	288.168	221.520

* With the integration of the proposed constrained search-tactic.

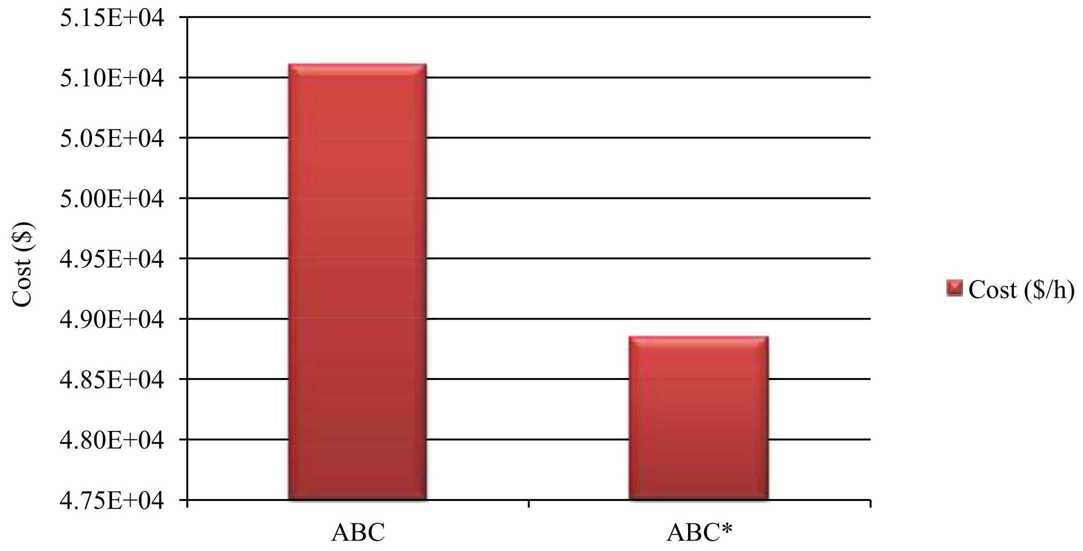


Figure 5. 6: Operating fuel costs for case 3-A due to ABC algorithm with and without the integration of the proposed constrained search-tactic.

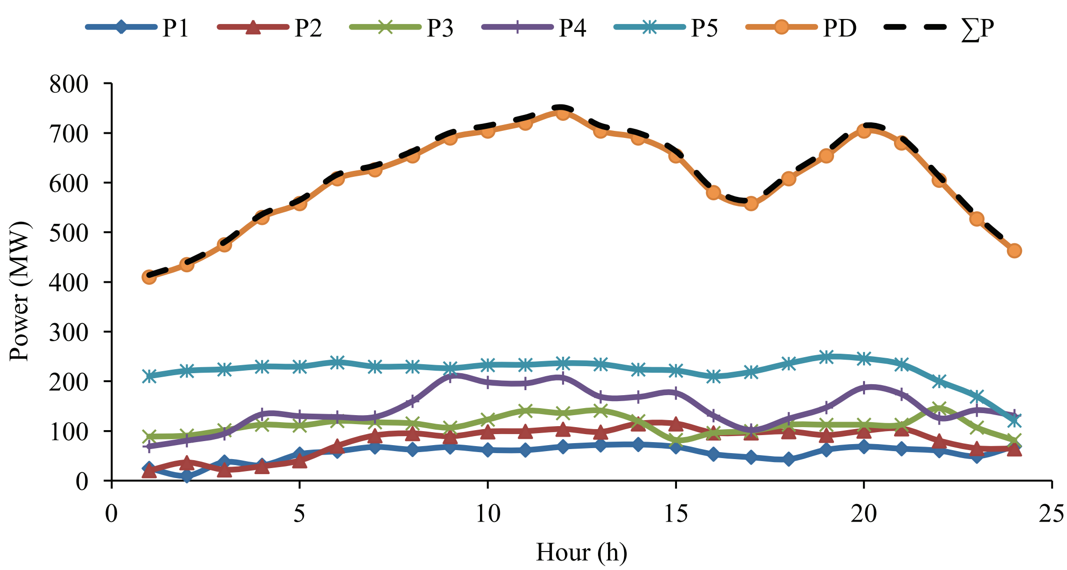


Figure 5. 7: Optimal units' dispatch schedule, load demand curve, and total power supply for case 3-A.

Table 5. 6: Optimal dispatch power for case 3-A using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch time (h)	Units' optimal output power P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	24.90640	20.44230	89.00000	69.00000	210.33820	413.68690	3.68681	0.00009
2	10.12730	36.44830	90.92960	80.64500	221.00020	439.15040	4.15043	0.00003
3	37.63000	22.00010	101.70000	94.48800	224.03600	479.85410	4.85418	0.00008
4	31.17770	28.85390	112.65200	133.75600	229.52000	535.95960	5.95962	0.00002
5	53.67679	40.35030	111.01879	130.01300	229.52000	564.57888	6.57897	0.00009
6	59.46100	70.00000	120.16420	128.20410	237.96910	615.79840	7.79841	0.00001
7	68.10929	90.79000	117.64000	128.21195	229.52000	634.27124	8.27126	0.00002
8	62.99900	95.05200	115.43330	160.03000	229.52000	663.03430	9.03425	0.00005
9	68.00000	89.09100	106.90000	210.00000	226.16150	700.15250	10.15240	0.00010
10	62.15450	98.54000	123.00350	198.01000	232.78100	714.48900	10.48900	0.00000
11	61.80400	99.54000	140.59600	195.87000	233.08070	730.89070	10.89080	0.00010
12	68.58800	103.76350	135.92940	206.97000	236.30170	751.55260	11.55260	0.00000
13	72.12000	98.00810	141.00000	168.87110	234.38000	714.37920	10.37920	0.00000
14	73.03300	114.53950	120.01190	168.60430	223.87860	700.06730	10.06730	0.00000
15	68.43499	114.54150	82.20120	176.48584	221.58730	663.25083	9.25088	0.00005
16	53.50430	96.40200	96.12900	131.11160	210.00000	587.14690	7.14700	0.00010
17	47.31110	96.45100	100.02000	102.10090	218.77329	564.65629	6.65635	0.00006
18	43.57430	98.89896	112.67330	124.91000	235.81870	615.87526	7.87530	0.00004
19	62.43000	91.54000	112.66000	147.31700	249.14970	663.09670	9.09669	0.00001
20	68.80430	100.02940	112.67330	187.32000	245.72590	714.55290	10.55290	0.00000
21	64.42430	104.60303	112.67310	174.06160	234.05680	689.81883	9.81892	0.00009
22	60.60000	80.30000	145.66000	126.01700	200.00000	612.57700	7.57693	0.00007
23	49.38430	65.02312	107.03100	142.02000	169.30200	532.76042	5.76039	0.00003
24	69.36430	64.35970	81.67300	131.06820	121.00143	467.46663	4.46665	0.00002
	Corresponding total operating fuel's emission (lb)				23,413.90			
	Total operating fuel's cost ($)				**48,848.20**			
	Total system's power loss (%)				1.300			

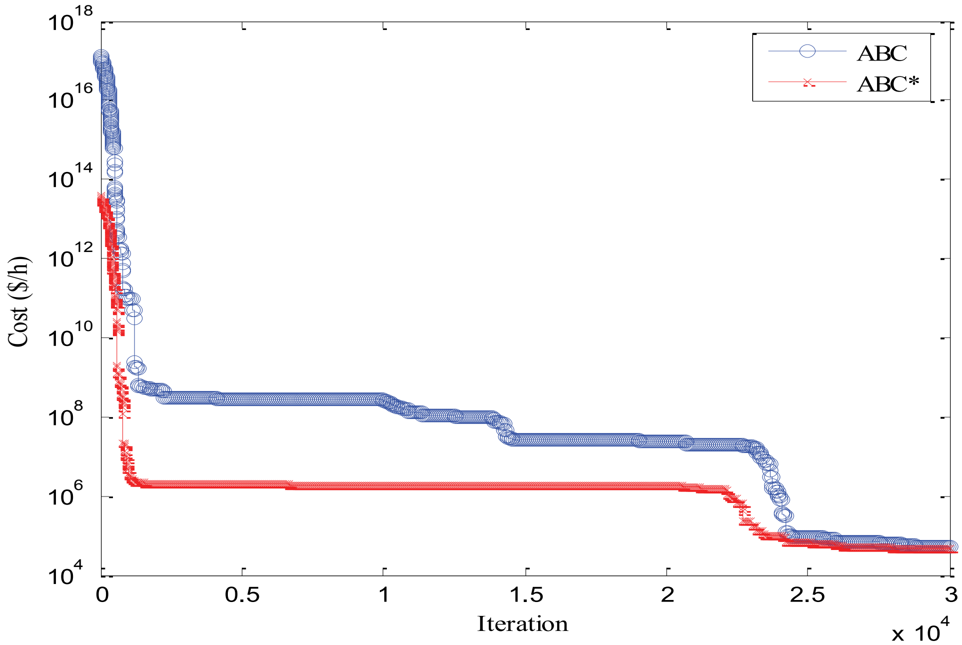


Figure 5. 8: The ABC algorithm's performance for case 3-A.

5.4.1.4 CASE 4-A

This system is a replica of that used in case 3-A except for the considered objective function. The objective was to minimize the on-line units' fuel emissions to meet the 24-hour load demand – subject to various constraints. The ABC algorithm's parameters are adjusted as in case 3-A, taking into account the test cases' dissimilarity in objective functions.

Once more, the proposed constrained search-tactic improved the outcomes of the ABC algorithm. An average of a 20% reduction in the required CPU time as well as a 0.6% reduction in the operating fuel's emissions was obtained by the strategy offered, as shown in Table 5.7 and Figure 5.9, respectively. As in Table 5.8, the optimal output power of the committed units satisfied the considered constraints. The output power of each unit in every time interval did not violate the unit's bound and ramp-rate limits. Furthermore, the summation of the units' output power in one time interval matched the power balance equation of the system. The system's load demand curve, total input power, and the units' dispatch schedules are illustrated in Figure 5.10. The ABC algorithm's performance for this case is demonstrated in Figure 5.11.

Table 5. 7: Comparison of results of the proposed ABC algorithm for case 4-A; Max: Maximum; Avg.: Average; Min: Minimum; Corresp.: Corresponding cost to the min (lb); Std.Dev.: Standard Deviation.

Method	Max. (lb)	Avg. (lb)	Min. (lb)	Corresp. ($)	Std.Dev.	CPU (s)
ABC	19,396.80	19,179.56	18,933.40	52,911.00	134.118	254.220
ABC*	19,716.00	19,179.95	**18,820.40**	53,166.50	170.375	203.220

* With the integration of the proposed constrained search-tactic.

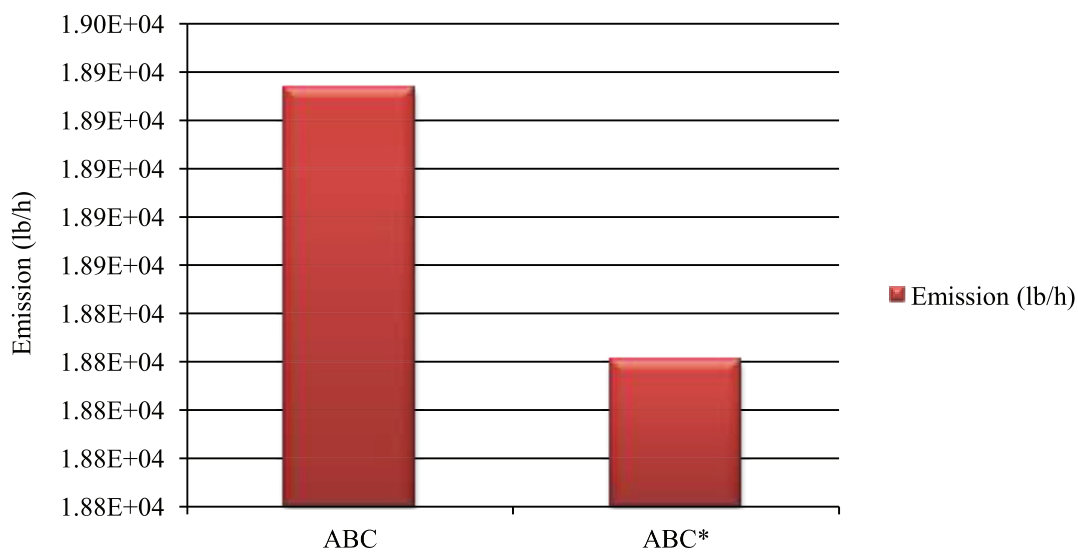


Figure 5. 9: Operating fuel emissions for case 4-A due to ABC algorithm with and without the integration of the proposed constrained search-tactic.

Table 5. 8: Optimal dispatch power for case 4-A using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch time (h)	Units' optimal output power P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	74.88750	48.47150	99.90000	71.13560	119.08200	413.47660	3.47657	0.00003
2	56.65000	53.63000	132.16900	80.73800	115.68450	438.87150	3.87142	0.00008
3	73.22300	67.74000	131.88000	129.00000	77.81470	479.65770	4.65773	0.00003
4	68.79510	90.00793	167.97681	103.96685	105.07000	535.81669	5.81674	0.00005
5	70.19149	75.11510	144.32602	121.11001	153.64510	564.38772	6.38767	0.00005
6	74.98991	88.60300	163.88894	128.42810	159.69400	615.60395	7.60392	0.00003
7	74.97850	115.41180	162.31620	143.05100	138.37010	634.12760	8.12762	0.00002
8	73.93760	92.03980	158.89720	161.35150	176.59500	662.82110	8.82103	0.00007
9	74.97171	120.79920	152.92030	175.62292	175.61080	699.92493	9.92495	0.00002
10	74.48694	121.85290	173.14902	197.04215	147.81380	714.34481	10.34490	0.00009
11	74.99012	97.01496	159.36030	213.22661	186.20610	730.79809	10.79810	0.00001
12	74.99533	124.66444	166.62258	163.85165	221.30210	751.43610	11.43610	0.00000
13	74.99686	124.08470	165.14000	177.96497	172.13400	714.32053	10.32060	0.00007
14	74.70000	116.09710	125.60000	210.43000	173.24270	700.06980	10.06990	0.00010
15	68.96000	86.32230	151.87234	160.63261	195.05198	662.83923	8.83926	0.00003
16	72.23230	57.30000	169.00000	137.00730	151.34000	586.87960	6.87960	0.00000
17	74.97350	75.56091	145.81569	135.36755	132.66633	564.38398	6.38398	0.00000
18	74.93450	64.50000	161.91290	169.44215	144.79760	615.58715	7.58717	0.00002
19	74.31327	93.43000	141.69900	158.89430	194.53634	662.87291	8.87290	0.00001
20	74.95600	95.80000	174.99534	164.29654	204.19220	714.24008	10.24010	0.00002
21	74.48384	122.73000	174.95392	159.89240	157.54250	689.60266	9.60269	0.00003
22	73.57050	97.49195	153.11980	178.89460	109.54550	612.62235	7.62238	0.00003
23	57.55440	70.16000	119.60000	131.00000	154.39830	532.71270	5.71269	0.00001
24	62.60000	50.30000	130.32000	105.24960	118.89824	467.36784	4.36781	0.00003
	Total operating fuel's emission (lb)				**18,820.40**			
	Corresponding total operating fuel's cost ($)				53,166.50			
	Total system's power loss (%)				1.272			

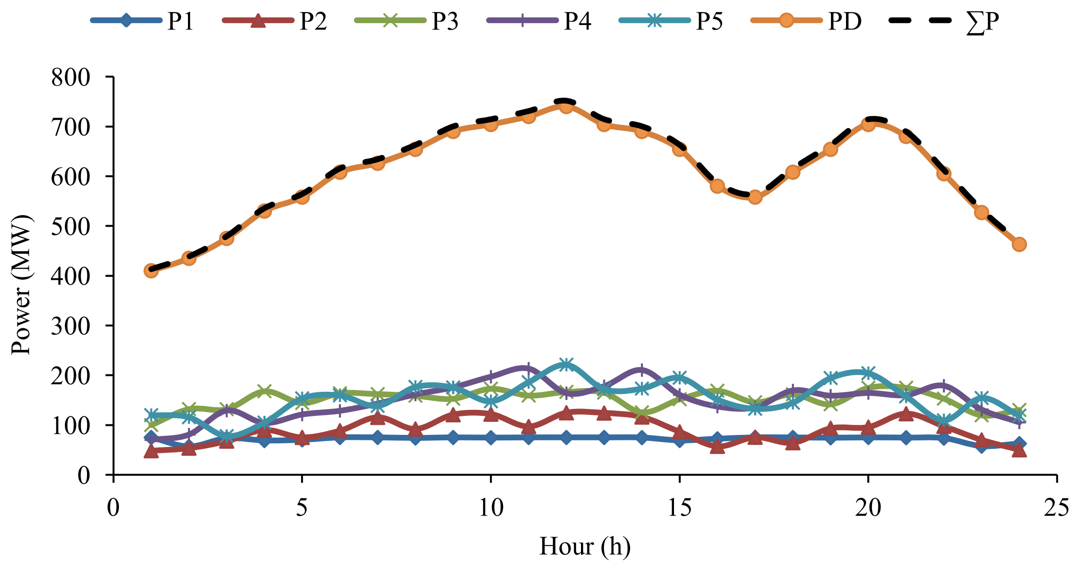


Figure 5. 10: Optimal units' dispatch schedule, load demand curve, and total power supply for case 4-A.

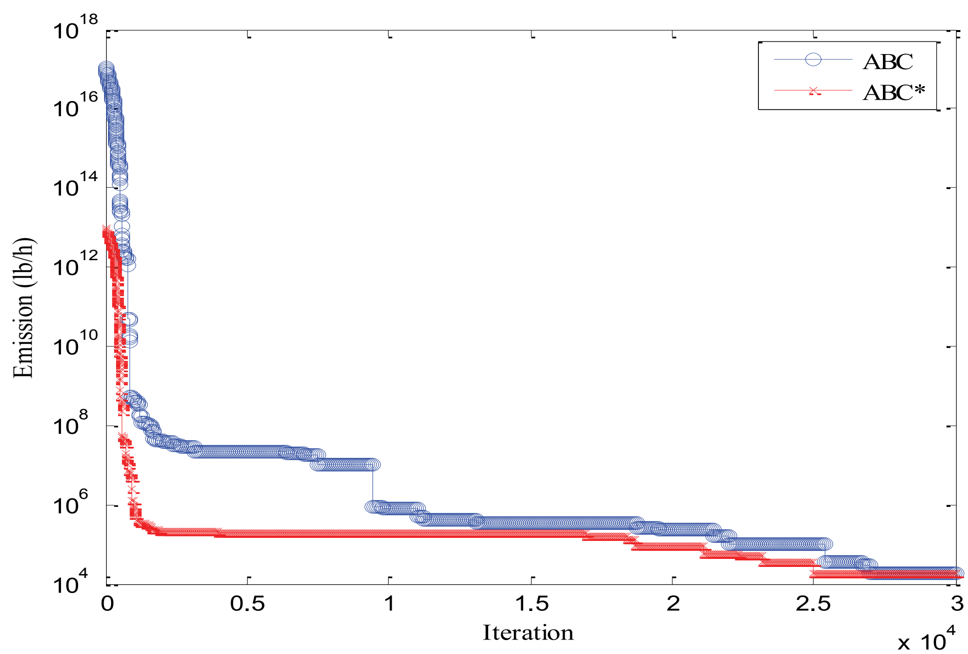


Figure 5. 11: The ABC algorithm's performance for case 4-A.

5.4.2 MULTIPLE OBJECTIVE FUNCTIONS PROBLEM

The DEED problem aims to minimize the operating fuel's cost and emission simultaneously, and to satisfy various practical constraints to meet the 24-hour load demand. In addition, the impact of integrating renewable sources (RS) on both fuel costs and emissions is examined in this set of tests. Because the effectiveness of the proposed constrained search-tactic has been proven previously, the ABC algorithm with the utilization of that designated tactic is employed in this set of test cases. The first test case solves the DEED problem without any contribution from an RS application. The subsequent cases are used to analyze the impact of an RS application on both objective functions. The contribution of the RS application will be based on percentages of the load demand, as reported in Table A.8, Table A.9, and Table A.10.

Although RSs, e.g., wind, tidal, and photovoltaic (PV), are environmentally-friendly practices, the intermittent nature of RS choices degrades their applicability as dispatchable options [160]. Despite that, the installation of wind power farms, for instance, is anticipated to supply 20% of the US demand by 2030 [165]. According to the Global Wind Energy Council (GWEC) [166], the installed capacity of wind power farms is increasing exponentially, as Figure 5.12 indicates.

The combined objective functions can be represented as a single objective function by linear scalar interpretation [159]. Hence, the mathematical objective function of the DEED problem becomes as follows:

$$\boldsymbol{min}\ \boldsymbol{\Psi} = w\ f_i(P_i) + \zeta_i(1-w)\ E_i(P_i) \tag{5.10}$$

where, $\boldsymbol{\Psi}$ is a combination of fuel costs and emissions objective functions, w is a varying weighting value, and ζ_i is the price associated with the emissions, where in this thesis ζ_i is set to one [167].

Competing functions have more than one optimal solution; thus, the Pareto-optimal solutions are attained by varying the w-value allowing the decision-maker to select the desired solution. The multiple objective functions (5.10) are in favor of only the cost minimization when the w-value equals one. In contrast, disregarding the influence of the cost and considering the emission minimization happens when the *w*-value sets to zero.

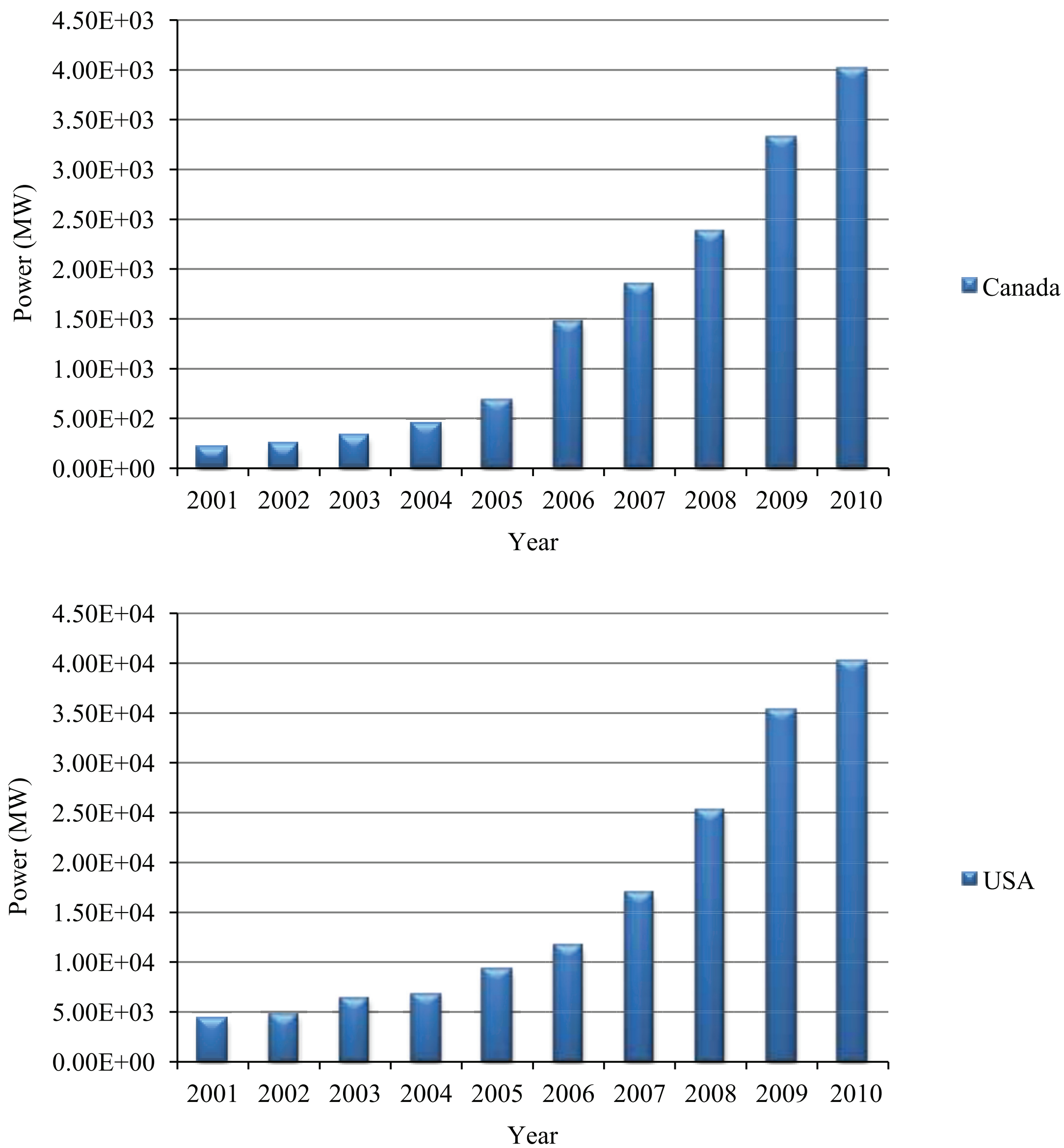


Figure 5. 12: Total installed capacity of wind power in North America.

5.4.2.1 CASE 1-B

The considered system, as shown in Table A.5, consists of five units. This system is identical to those in case 3-A and case 4-A. However, the objective function at this point, and hereafter, was to minimize simultaneously both the operating fuel's cost and emission of committed generating units. Therefore, the trade-off curve between costs and emissions will be demonstrated based on varying the *w* factor used in the linear scalar expression (5.10). The *w*-values in this thesis are selected as: 0.00, 0.20, 0.50, 0.80, and 1.00. The ABC* parameters for this system were adjusted as: 300, 18×10^3, and 30×10^3 for *CS*, *limit*, and *MCN*, respectively. Minimizing the system's power balance mismatch will be (temporarily) the target until the 3,001 (φ+1) cycle starts. Afterward, the problem's main objective function will be considered.

The summarized results of the proposed ABC* algorithm successfully captured the Pareto-optimal shape as presented in Table 5.9. The trade-off curve between the cost and emission for this system is exemplified in Figure 5.13. As recorded in Table 5.9, the corresponding results at *w*-values' one and zero were identical to those attained in case 3-A and case 4-A, respectively. Accordingly, further confirmation of the efficiency of the proposed ABC algorithm and the constrained search-tactic was shown. It is important to mention that there is no single optimal solution in such a scenario where multiple objective functions are considered. Therefore, the decision-maker must select the desired solution based on his/her preference. These preferences vary from one utility to another and from one country to another. Political, social, and economical factors play key roles in shifting those preferences.

The result of the proposed ABC* algorithm at a 0.50 *w*-value was compared with those obtained by PSO [31] and PS [44]. The summarized results are reported in Table 5.10. Although both PSO and PS methods attained "less" operating fuel costs than that of the ABC* algorithm, they disregarded the P^0 scheduling values, and relaxed the accepted value for violating the equality constraints. The dispatch schedule with a high power output mismatch degraded the practicality of the attainable solutions by these methods. Clearly, as shown in Figure 5.14, the dispatch schedule of on-line units in every time interval was more consistent using the ABC* algorithm than the compared methods. The proposed ABC* algorithm provided the least absolute value of violating the system's equality constraints and, therefore, represented the minimum total system's power loss. As indicated in Table 5.11, this added practical value to the presented result. Furthermore, the results of the ABC* method outperformed those found by the PSO method in terms of operating fuel's emission. The optimal dispatch schedule for this system by the proposed ABC* algorithm is recorded in Table 5.11 with a 50% weight for each objective function. In addition, the optimal units' scheduling by the ABC* algorithm in a two-dimension format is exemplified in Figure 5.15.

Although relaxing the equality constraints' violation plays a significant role (in addition to other factors such as the utilized algorithm and PC's features) in achieving a faster solution, the offered ABC* algorithm outperformed that of PS method with respect to the CPU time requirement (~60%) as seen in Table 5.10. On the other hand, only 7.71% reduction in the CPU time was due to PSO.

Table 5. 9: Summarized results of the proposed ABC* algorithm for case 1-B.

w-value	Emission (lb)	Cost ($)	Total output power (MW)	Total system's power loss (%)
1.00	23,413.90	48,848.20	14,769.06688	1.300
0.80	20,229.40	51,169.40	14,765.45212	1.276
0.50	19,661.40	51,403.90	14,765.01644	1.273
0.20	19,205.80	52,580.90	14,764.80358	1.272
0.00	18,820.40	53,166.50	14,764.76562	1.272

* With the integration of the proposed constrained search-tactic.

Table 5. 10: Comparison of results of the proposed ABC algorithm at 0.50 *w*-value for case 1-B.

Method	Emission (lb)	Cost ($)	Total system's power loss (%)	Average CPU time (s)
ABC*	19,661.40	51,403.90	1.273	206.10
PSO [31]	20,163.00	50,893.00	1.303	190.20
PS [44]	18,927.00	47,911.00	1.320	514.25

* With the integration of the proposed constrained search-tactic.

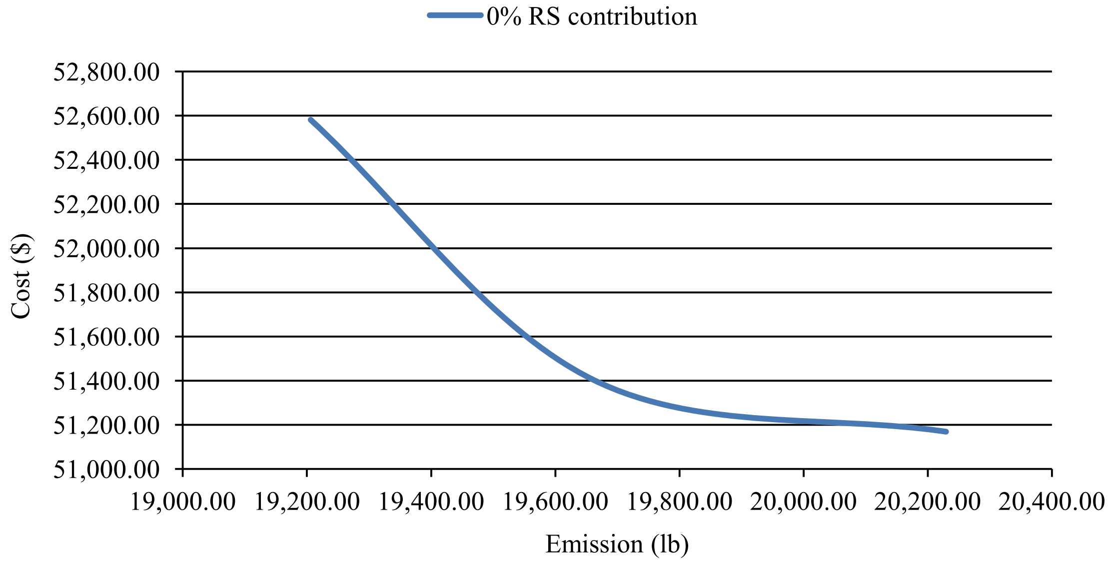


Figure 5. 13: Trade-off curve between the two objective functions of case 1-B.

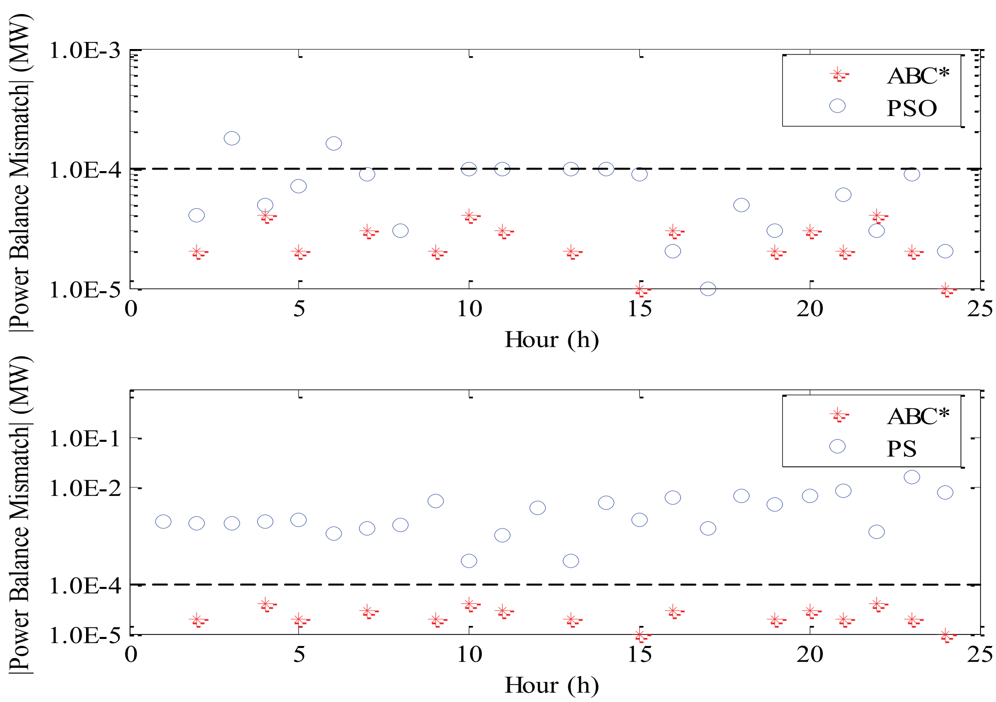


Figure 5. 14: Comparison of the system's equality constraints' violations at every dispatched period due to different methods for case 1-B.

Table 5. 11: Optimal dispatch power for case 1-B at 0.50 *w*-value using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch	Units' optimal output power							
time (h)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	74.99990	48.47890	100.00040	70.00000	120.00000	413.47920	3.47920	0.00000
2	62.62190	52.80150	74.80550	92.49451	156.24971	438.97312	3.97314	0.00002
3	52.78129	69.65155	106.24417	124.08349	126.87801	479.63851	4.63851	0.00000
4	74.56080	83.37560	102.97240	125.86510	149.06150	535.83540	5.83544	0.00004
5	72.72465	89.08120	137.88230	155.38635	109.37079	564.44529	6.44531	0.00002
6	74.99626	106.87343	173.12195	125.00948	135.64819	615.64931	7.64931	0.00000
7	69.63258	100.52984	167.84744	135.88547	160.18611	634.08144	8.08147	0.00003
8	74.93228	97.74046	174.99965	134.97654	180.17650	662.82543	8.82543	0.00000
9	74.89880	105.28200	174.99000	132.46900	212.24840	699.88820	9.88822	0.00002
10	74.55611	96.70946	174.33870	159.52942	209.11567	714.24936	10.24940	0.00004
11	74.99690	103.33726	174.99480	139.31058	238.16853	730.80807	10.80810	0.00003
12	73.96900	94.08280	160.67510	188.71050	233.97000	751.40740	11.40740	0.00000
13	74.95150	123.58509	141.69911	164.13981	210.01907	714.39458	10.39460	0.00002
14	74.53264	103.09389	161.58785	146.87578	213.79895	699.88911	9.88911	0.00000
15	74.43417	114.40548	150.25680	117.54962	206.30187	662.94794	8.94793	0.00001
16	69.38033	89.03049	113.25510	154.16410	161.17065	587.00067	7.00070	0.00003
17	70.84158	87.54273	133.28219	106.19473	166.58076	564.44199	6.44199	0.00000
18	70.01122	98.29720	137.17190	134.36076	175.81794	615.65902	7.65902	0.00000
19	74.99588	92.79690	147.68767	151.74886	195.62864	662.85795	8.85797	0.00002
20	74.82567	118.07610	170.91482	153.84448	196.63250	714.29357	10.29360	0.00003
21	73.99180	120.20180	174.99820	166.64818	153.75987	689.59985	9.59987	0.00002
22	74.99570	90.74260	174.51890	126.65483	145.62133	612.53336	7.53340	0.00004
23	74.54800	64.29660	140.74650	128.99550	124.09549	532.68209	5.68211	0.00002
24	54.46970	75.12080	111.75558	80.75600	145.33350	467.43558	4.43559	0.00001
	Total operating fuel's emission (lb)				19,661.40			
	Total operating fuel's cost ($)				51,403.90			
	Total system's power loss (%)				1.273			

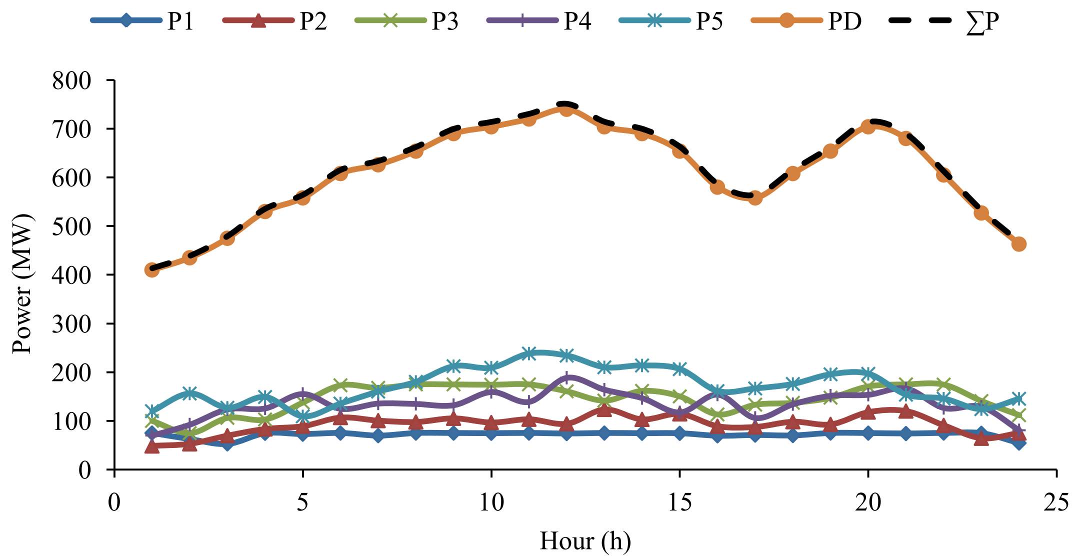


Figure 5. 15: Optimal units' dispatch schedule, load demand curve, and total power supply for case 1-B at 0.50 *w*-value; with non-RS contribution.

5.4.2.2 CASE 2-B

The impact of integrating renewable energy resources was examined in this case. A wind power farm designated to share a 10% of the system's load demand. For this test case and the subsequent ones, the ABC algorithm's parameters were tuned as in case 1-B.

As shown in Table 5.12, and with respect to Table 5.9, the integration of RS decreased the operating fuel's cost and emission as well as the system's total real power loss. As of an evenly distributed *w*-values' perspective, the 10% contribution of a RS led to 16.63% and 6.56% reduction in the fuel's emission and cost, respectively, for this system with regard to non-RS integration scenario. The total system's loss was decreased by 18.96% due to the 10% sharing of RS practice, as well. The average CPU time required to attain a solution in this case was approximately 200.030 seconds, which was slightly less than that of the previous test case.

The ABC* algorithm successfully captured the Pareto-optimal curve for this test case, as demonstrated in Figure 5.16. Furthermore, the optimal dispatch of the committed units, as in Table 5.13, confirmed the effectiveness of the offered technique by satisfying the problem's equality and inequality constraints. The units' output power and RS contribution to meet the load demand curve are illustrated in Figure 5.17. The gap between the demand and units' total power input curves is covered by the RS share as in Figure 5.17.

Table 5. 12: Summarized results of the proposed ABC* algorithm for case 2-B.

w-value	Emission (lb)	Cost ($)	Total output power (MW)	Total system's power loss (%)
1.00	17,449.80	47,522.60	13,272.66390	1.155
0.80	17,141.60	47,812.50	13,272.32621	1.153
0.50	16,390.90	48,031.90	13,271.67395	1.148
0.20	16,065.20	48,616.90	13,271.43784	1.146
0.00	15,848.60	48,736.20	13,270.96116	1.143

* With the integration of the proposed constrained search-tactic.

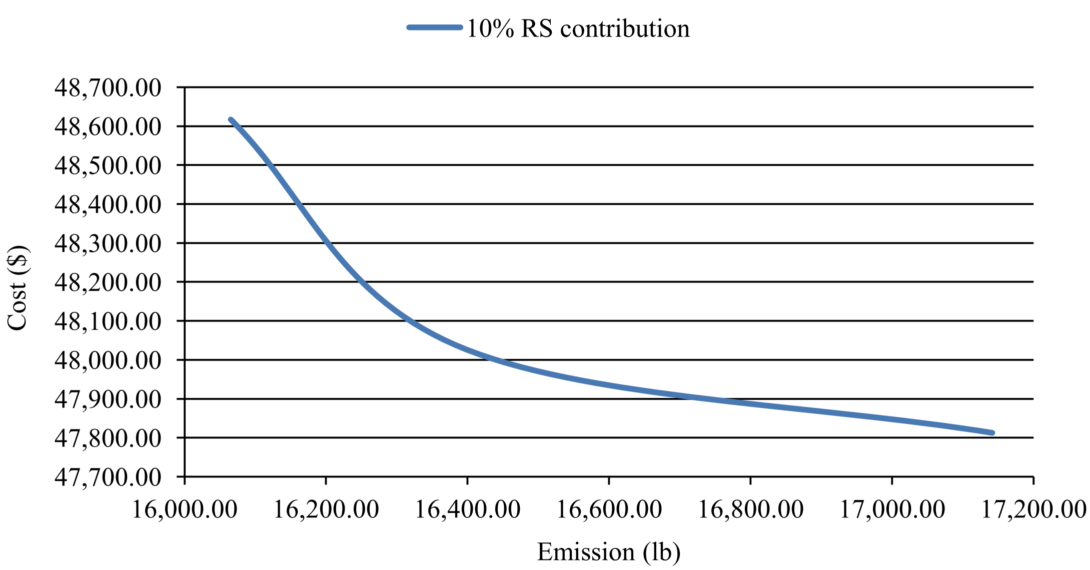


Figure 5. 16: Trade-off curve between the two objective functions of case 2-B.

Table 5. 13: Optimal dispatch power for case 2-B at 0.50 *w*-value using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch time (h)	Units' optimal output power P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	53.00000	35.00000	80.00000	78.00000	125.80800	371.80800	2.80800	0.00000
2	41.52671	22.72072	59.04247	112.17445	159.30904	394.77339	3.27339	0.00000
3	54.04531	32.82272	72.89724	125.78457	145.77844	431.32828	3.82829	0.00001
4	55.30327	61.28090	111.90429	140.43840	112.75023	481.67709	4.67699	0.00010
5	73.75176	71.01385	146.82394	104.63467	111.15078	507.37500	5.17505	0.00005
6	74.83398	61.32709	149.57931	118.62269	148.96714	553.33021	6.13017	0.00004
7	74.54523	86.20447	121.66814	120.95956	166.59733	569.97473	6.57476	0.00003
8	73.45460	80.06769	117.26042	136.72939	188.29591	595.80801	7.20803	0.00002
9	73.75545	52.59164	137.28443	149.71515	215.66379	629.01046	8.01044	0.00002
10	74.03425	82.54305	174.47615	134.30462	176.49595	641.85402	8.25399	0.00003
11	74.79112	86.32927	166.35420	135.10758	194.08073	656.66290	8.66294	0.00004
12	74.68600	101.73674	174.89310	177.26842	146.58707	675.17133	9.17134	0.00001
13	74.50062	98.54222	155.82075	130.30378	182.73537	641.90274	8.30277	0.00003
14	74.77300	90.29784	123.51991	128.10537	212.38163	629.07775	8.07765	0.00010
15	73.30716	90.57579	86.07161	152.46431	193.54286	595.96173	7.36177	0.00004
16	74.78575	63.93380	111.02042	127.30997	150.56934	527.61928	5.61936	0.00008
17	72.61711	57.71420	113.02619	100.59487	163.46343	507.41580	5.21586	0.00006
18	74.82271	85.50012	98.65364	132.50042	161.97683	553.45372	6.25367	0.00005
19	74.42990	75.30056	124.66038	146.01526	175.36020	595.76630	7.16639	0.00009
20	74.84783	84.86916	164.46330	151.11586	166.55457	641.85072	8.25072	0.00000
21	74.86557	101.33777	139.56298	131.41717	172.57875	619.76224	7.76226	0.00002
22	74.99402	88.09683	119.92637	127.99268	139.61607	550.62597	6.12595	0.00002
23	50.91754	66.73125	80.67614	111.22569	169.47588	479.02650	4.72654	0.00004
24	51.46155	44.12023	48.01298	127.56253	149.28049	420.43778	3.73776	0.00002
	Total operating fuel's emission (lb)				16,390.90			
	Total operating fuel's cost ($)				48,031.90			
	Total system's power loss (%)				1.148			

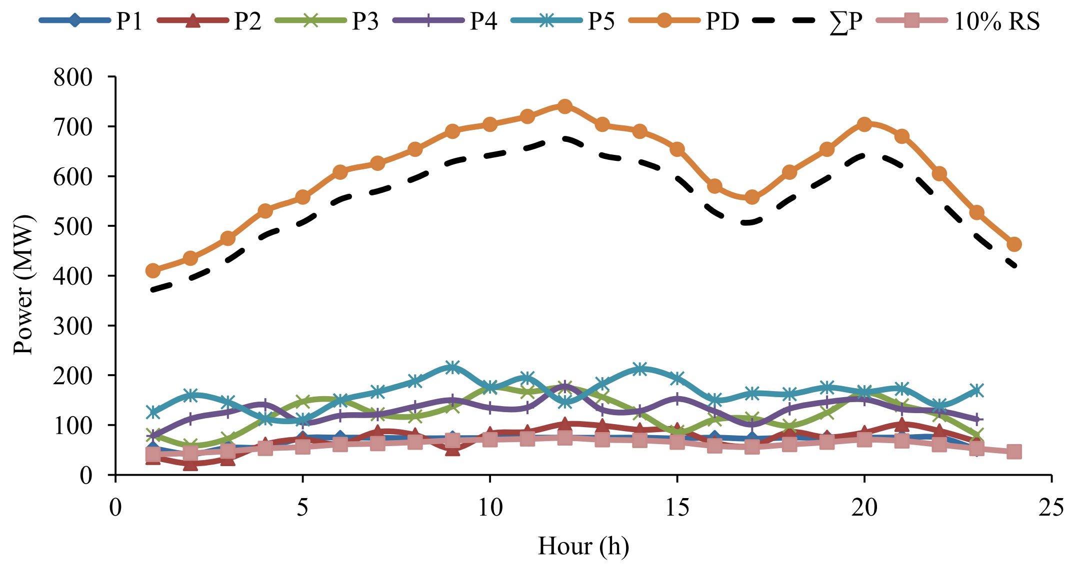


Figure 5. 17: Optimal units' dispatch schedule, load demand curve, and total power supply for case 2-B at 0.50 *w*-value; with a 10% RS contribution.

5.4.2.3 CASE 3-B

The contribution of the RS practice in this test increased by a 5% with respect to case 2-B, resulted in a designated wind power farm supplying 15% of the load demand.

As listed in Table 5.14, the 5% increase of the RS contribution drove this system to attain extra reductions in the operating fuel's cost (4.26%) and emission (10.36%) with regard to case 2-B from a 0.50 *w*-value standpoint. In addition, the system's total real power loss decreased by 8.60% due to the additional 5% of RS supply. The offered ABC algorithm required approximately 199.870 seconds to obtain the solution for this test case.

The proposed ABC algorithm, as shown in Figure 5.18, sufficiently achieved the trade-off curve between the conflicting objective's functions considered in this test case. Furthermore, the influence of the suggested constrained search-tactic led the ABC method to obtain the optimal output power dispatch of on-line units without violating the considered constraints. As in Table 5.15, every unit's output power at each time interval was consistent with the constraints associated with that unit and the neighbouring ones. In addition, the system's power balance constraint was less than 10^{-4}, i.e., the acceptable range, at every time period. To meet the load demand at every hour, Figure 5.19 demonstrates the output power of each unit as well as the RS contribution per time interval. The increase in the gap between the curves of the load demand and units' total input power was due to the 5% increase of the RS's share with respect to the previous case.

Table 5. 14: Summarized results of the proposed ABC* algorithm for case 3-B.

w-value	Emission (lb)	Cost ($)	Total output power (MW)	Total system's power loss (%)
1.00	15,039.70	45,773.40	12,527.04666	1.090
0.80	14,737.50	45,795.60	12,526.35244	1.085
0.50	14,353.10	45,843.80	12,526.65636	1.087
0.20	14,237.90	46,032.40	12,525.98158	1.082
0.00	14,138.70	47,191.80	12,525.95154	1.082

* With the integration of the proposed constrained search-tactic.

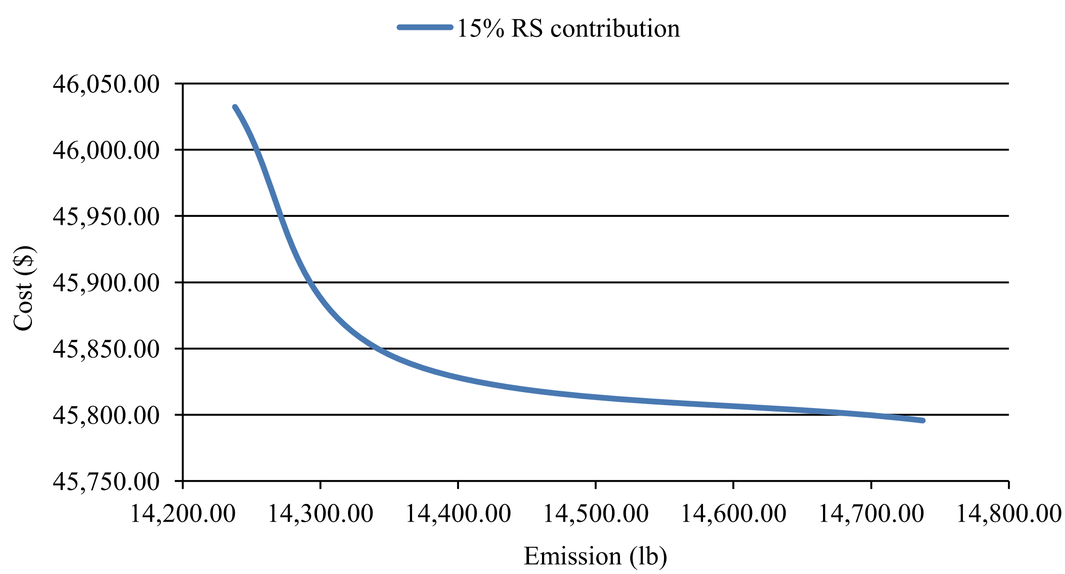


Figure 5. 18: Trade-off curve between the two objective functions of case 3-B.

Table 5. 15: Optimal dispatch power for case 3-B at 0.50 *w*-value using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch	Units' optimal output power							
time (h)	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	58.50000	36.04000	80.01030	59.90020	116.56680	351.01730	2.51730	0.00000
2	49.17548	40.43493	97.64108	87.67862	97.59536	372.52547	2.77554	0.00007
3	49.05515	48.90092	108.63515	61.15520	139.37933	407.12575	3.37576	0.00001
4	73.39811	76.66488	82.80997	106.60703	115.24619	454.72618	4.22613	0.00005
5	74.77022	87.10783	97.23428	119.36858	100.50010	478.98101	4.68106	0.00005
6	74.73358	84.46878	108.40610	128.69511	126.02507	522.32864	5.52863	0.00001
7	74.58555	114.15753	130.79596	88.42772	130.05892	538.02568	5.92569	0.00001
8	73.69007	88.80206	129.01259	112.40389	158.37956	562.28817	6.38825	0.00008
9	74.86566	89.89091	114.40777	145.67283	168.82067	593.65784	7.15785	0.00001
10	71.83120	103.38105	125.12021	153.01141	152.50495	605.84882	7.44878	0.00004
11	70.75540	104.10951	119.79159	124.12058	201.08414	619.86122	7.86123	0.00001
12	71.80324	124.08285	133.32519	119.24951	188.84867	637.30946	8.30938	0.00008
13	73.08023	124.56459	128.56698	130.81257	148.87207	605.89644	7.49636	0.00008
14	74.99325	112.30418	152.30883	120.43861	133.59225	593.63712	7.13711	0.00001
15	74.99598	86.01356	114.51512	155.33623	131.45669	562.31758	6.41757	0.00001
16	60.72062	107.38082	105.84459	138.36594	85.81696	498.12893	5.12902	0.00009
17	74.53812	83.11926	85.80413	99.84836	135.69311	479.00298	4.70307	0.00009
18	67.09932	61.34563	91.60594	116.87887	185.47577	522.40553	5.60559	0.00006
19	74.17772	86.63257	130.20693	131.87000	139.37982	562.26704	6.36700	0.00004
20	64.77167	88.84877	153.35669	130.88337	167.91115	605.77165	7.37157	0.00008
21	74.57650	80.89549	125.05388	135.61963	168.76003	584.90553	6.90552	0.00001
22	68.47252	70.62993	111.76509	117.37083	151.46370	519.70207	5.45205	0.00002
23	74.57514	59.93987	102.36644	72.75499	142.48830	452.12474	4.17473	0.00001
24	70.65905	68.30037	81.89761	57.00616	118.93804	396.80123	3.25117	0.00006
	Total operating fuel's emission (lb)				14,353.10			
	Total operating fuel's cost ($)				45,843.80			
	Total system's power loss (%)				1.087			

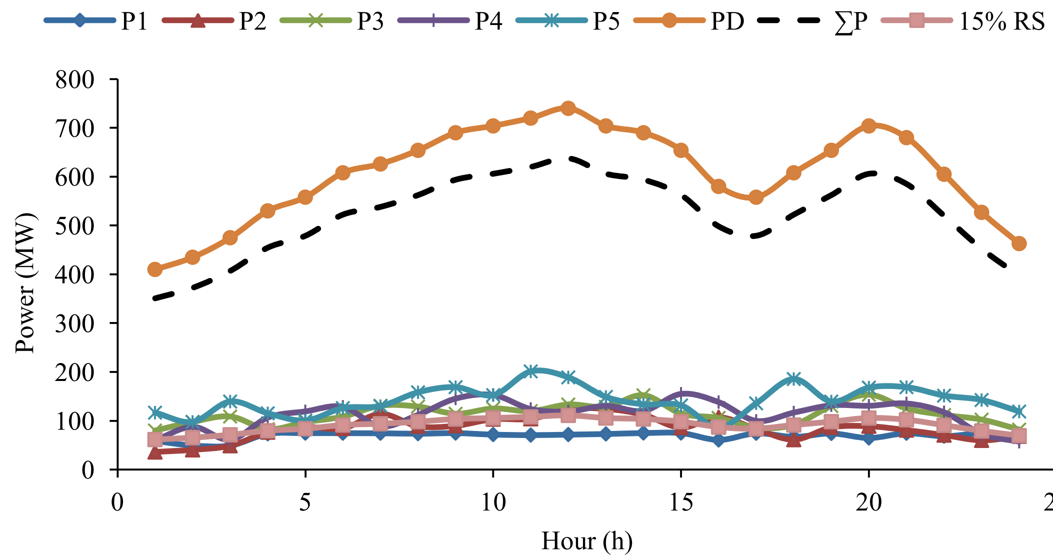


Figure 5. 19: Optimal units' dispatch schedule, load demand curve, and total power supply for case 3-B at 0.50 *w*-value; with a 15% RS contribution.

5.4.2.4 CASE 4-B

The contribution of the RS practice in this test was doubled in comparison with that of case 2-B. The wind power farm supplied a 20% of the load demand, and the parameters of the ABC algorithm adjusted identically to those of the preceding test cases. The reductions in the operating fuel's cost and emission, as recorded in Table 5.16, virtually doubled the revealed outcomes of a test case 2-B.

As summarized in Table 5.16, and according to the corresponding results of a 0.50 *w*-value, 33.95% and 13.25% were the additional reductions in the operating fuel's emission and cost respectively due to a 20% integration of wind power farm relating to a non-RS contribution scenario. This case increased the reductions of the fuel's cost and emission by 2.43% and 6.95% concerning the case 3-B. The enhancements between this case and case 3-B in terms of system's total real power loss and the required CPU time were 8.20% and 4.320 seconds correspondingly. Evidently, as exemplified in Figure 5.20, the proposed ABC algorithm successfully reached the Pareto-optimal relationship between the two competing (cost and emission) functions. The retained units' optimal dispatched output power due to a 0.50 *w*-value obtained by the ABC algorithm is shown in Table 5.17. Clearly, the attained results were satisfying the DEED problem's equality and inequality constraints. Furthermore, the small (less than 10^{-4}) power mismatch per time interval added further value to the reported solution. Obviously, as shown in Figure 5.21, the RS's supply curve was slightly higher than that of the case 3-B in Figure 5.19. This explained the larger gap between the total supplied power curve from the on-line units and demand curve in Figure 5.21 in contrast with Figure 5.19.

Table 5. 16: Summarized results of the proposed ABC* algorithm for case 4-B.

w-value	Emission (lb)	Cost ($)	Total output power (MW)	Total system's power loss (%)
1.00	13,662.90	44,387.10	11,782.38211	1.025
0.80	13,178.10	44,512.80	11,782.37390	1.025
0.50	12,986.90	44,593.60	11,782.39779	1.025
0.20	12,975.90	44,798.60	11,781.98556	1.022
0.00	12,936.70	45,260.50	11,781.99104	1.022

* With the integration of the proposed constrained search-tactic.

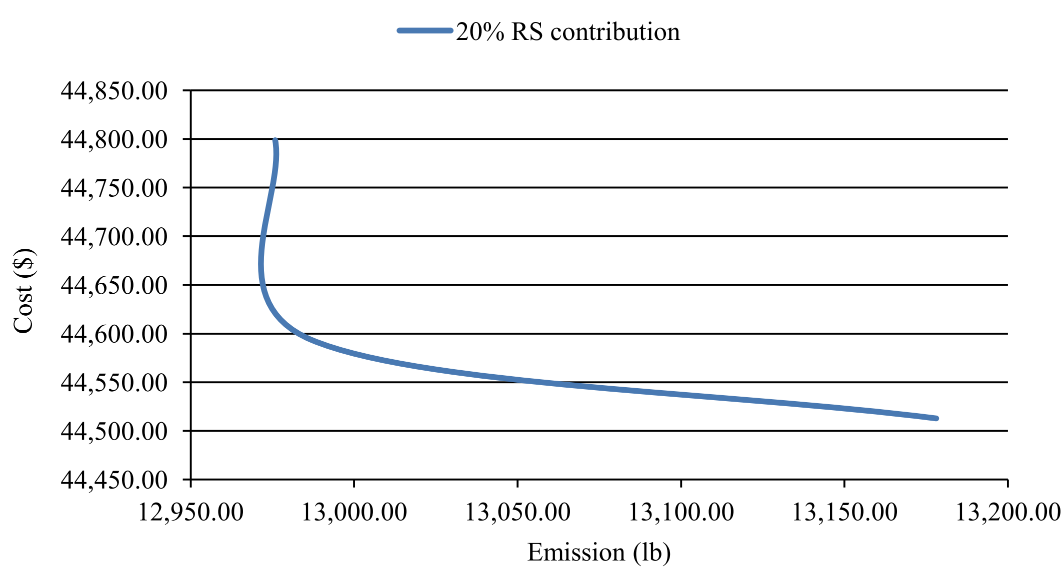


Figure 5. 20: Trade-off curve between the two objective functions of case 4-B.

Table 5. 17: Optimal dispatch power for case 4-B at 0.50 *w*-value using ABC algorithm with the integration of the proposed constrained search-tactic; Sys.Viol.: System's Power Balance Constraints' Violation.

Dispatch time (h)	Units' optimal output power P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	$\sum P$ (MW)	P_L (MW)	|Sys.Viol.| (MW)
1	48.24190	40.07970	70.00390	59.90000	112.00000	330.22550	2.22557	0.00007
2	67.03691	39.43458	93.57669	56.28241	94.17190	350.50249	2.50250	0.00001
3	38.02966	44.62586	94.13695	65.40121	140.79755	382.99123	2.99117	0.00006
4	52.20106	50.58517	111.61401	63.88970	149.44275	427.73269	3.73265	0.00004
5	53.72440	67.09989	140.41224	83.08241	106.17312	450.49206	4.09197	0.00009
6	72.19318	92.05107	132.27422	80.45414	114.33798	491.31059	4.91065	0.00006
7	74.99728	88.21582	107.28445	121.42398	114.07615	505.99768	5.19764	0.00004
8	74.47910	60.31100	119.62067	125.60410	148.80979	528.82466	5.62472	0.00006
9	73.26435	78.57084	131.15139	128.31565	146.96190	558.26413	6.26420	0.00007
10	74.98506	104.07879	123.27460	127.14993	140.29929	569.78767	6.58760	0.00007
11	68.90699	97.34105	139.88427	167.09027	109.68163	582.90421	6.90422	0.00001
12	74.16591	108.99040	111.56726	166.85371	137.78714	599.36442	7.36433	0.00009
13	65.34142	100.81883	116.46371	119.27922	167.90777	569.81095	6.61099	0.00004
14	59.29433	89.03314	121.97249	94.28200	193.80300	558.38496	6.38490	0.00006
15	73.38526	92.93562	110.78345	107.62488	144.15556	528.88477	5.68481	0.00004
16	68.67473	91.95737	72.21200	141.58800	94.15700	468.58910	4.58907	0.00003
17	68.46918	84.33368	60.41172	125.03205	112.39703	450.64366	4.24357	0.00009
18	56.35633	65.44752	81.50096	143.03957	145.01164	491.35602	4.95604	0.00002
19	46.26593	90.98770	119.36274	113.52709	158.73283	528.87629	5.67633	0.00004
20	71.12098	82.88318	106.92039	134.66419	174.21967	569.80841	6.60849	0.00008
21	74.98291	74.22404	85.24859	129.04818	186.74942	550.25314	6.25315	0.00001
22	72.24775	80.34933	91.10999	107.46050	137.70832	488.87589	4.87581	0.00008
23	50.00059	69.71443	90.21536	103.13704	112.19274	425.26016	3.66012	0.00004
24	35.54416	61.41116	73.74491	115.91891	86.63797	373.25711	2.85717	0.00006
	Total operating fuel's emission (lb)				12,986.90			
	Total operating fuel's cost ($)				44,593.60			
	Total system's power loss (%)				1.025			

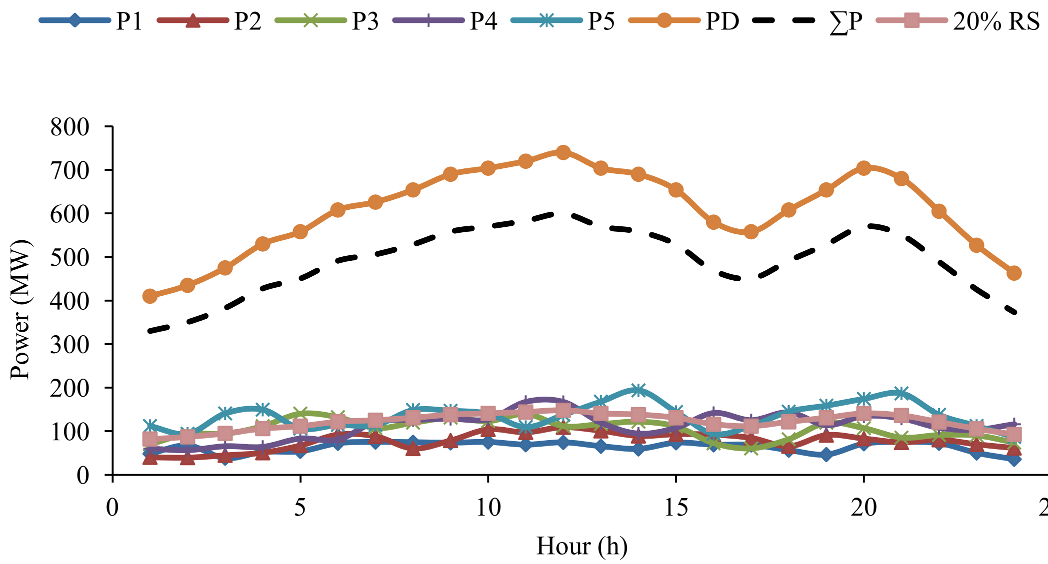


Figure 5. 21: Optimal units' dispatch schedule, load demand curve, and total power supply for case 4-B at 0.50 *w*-value; with a 20% RS contribution.

To analyze the effect of the selected RS applications, the outcomes at pure cost's minimization and emission's minimization are taken into account. In addition, the impact of the RS's integration at evenly distributed weights is considered. Although the RS's contribution has a positive impact on both objective functions, it increases the uncertainty of the overall system. One solution is to increase the system spinning reserve, but this would increase the operating cost in favor of the system's reliability. Units' coefficients and characteristics and load demand significantly influence the results, as they may vary from one system to another. Therefore, the conclusions are not necessarily applicable in other electric power systems. The decision-maker, undoubtedly, plays a key role in selecting which practice meets the desired aims.

From a single objective function perspective, the reductions of both competing functions increased as the contributions of the RS increased, as concluded in Table 5.18. The improvements in the operating fuel's cost and emission (individually) are demonstrated in Figure 5.22. Clearly, significant daily cost savings of \$1,325.60, \$3,074.80, and \$4,461.10 are the results of 10%, 15%, and 20% of the RS's sharing, respectively. Emission's pollutants (in lb) decline by 2,971.80, 4,681.70, and 5,883.70

due to these same contributions. The system's real power loss – for both functions – is also decreased by an average of 19.69%, 28.36%, and 36.50% because of the chosen RS practices.

While assigning equal weights to the competing objective functions, both gain positively from the selected RS applications, as exemplified in Figure 5.23. Considerable 24-hour cost savings of \$3,372.00, \$5,560.10, and \$6,810.30 are the results of the RS supplying 10%, 15%, and 20% of the load's demand. In addition, the NO_x emission – due to the same percentages of RS's integrations – is decreasing daily (in lb) by 3,270.50, 5,308.30, and 6,674.50. The system's real power loss is also reduced by 18.96%, 27.56%, and 35.75%.

Table 5. 18: Summarized (row by column) reductions in operating fuel's cost and emission attained by the proposed ABC* algorithm with different integration levels of the RS cases at one and zero *w*-values.

RS's Contribution	Objective Function	*RS's Contribution*			
		0% RS	*10% RS*	*15% RS*	*20% RS*
0% RS	Emission (lb)	**0.000%**	15.790%	24.876%	31.262%
	Cost (\$)	**0.000%**	2.714%	6.295%	9.133%
10% RS	Emission (lb)	-15.790%	**0.000%**	10.789%	18.373%
	Cost (\$)	-2.714%	**0.000%**	3.681%	3.681%
15% RS	Emission (lb)	-24.876%	-10.789%	**0.000%**	8.501%
	Cost (\$)	-6.295%	-3.681%	**0.000%**	3.029%
20% RS	Emission (lb)	-31.262%	-18.373%	-8.501%	**0.000%**
	Cost (\$)	-9.133%	-6.598%	-3.029%	**0.000%**

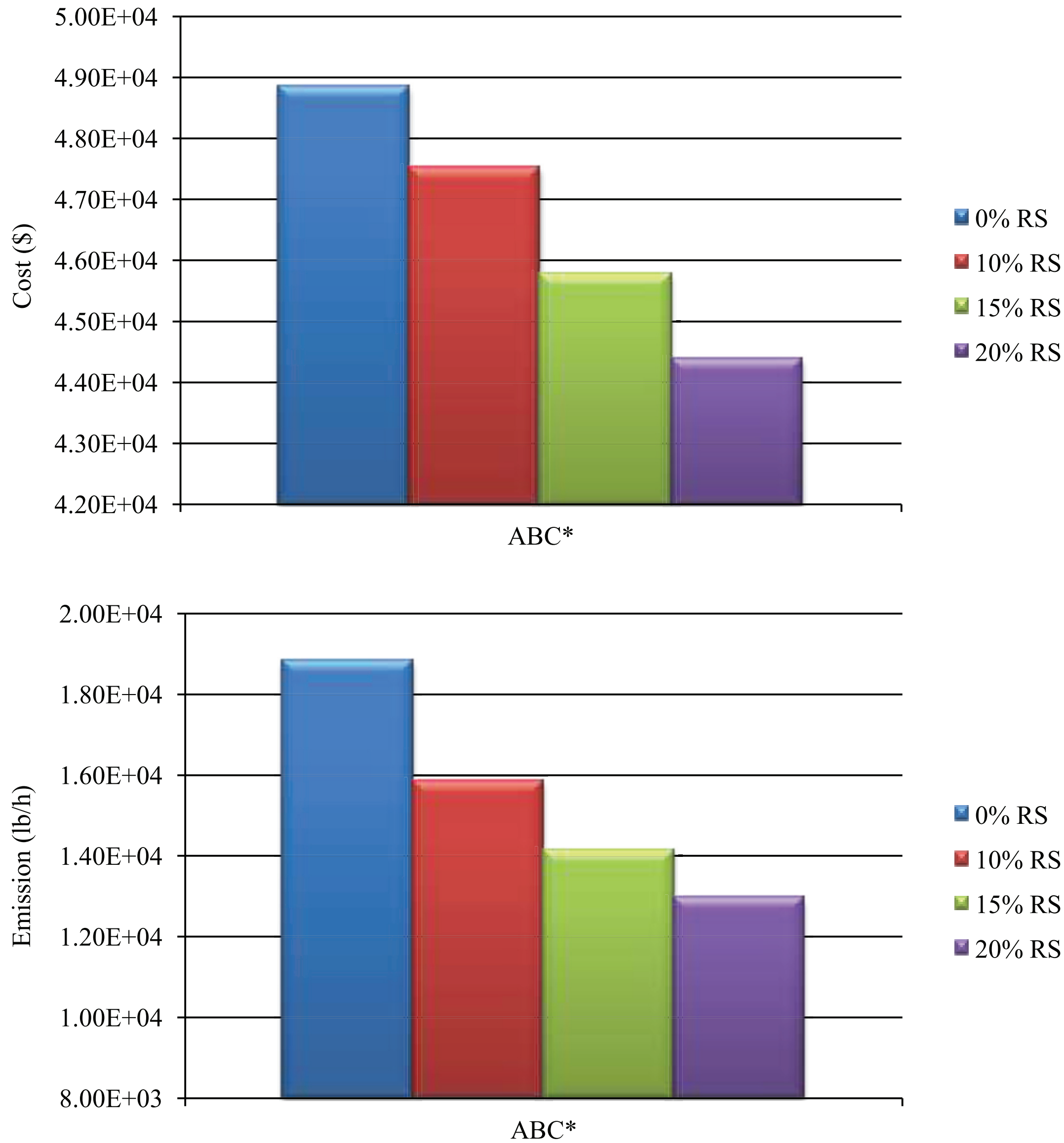


Figure 5. 22: Operating fuel's cost and emission reductions due to different RS contribution's levels obtained by the ABC* algorithm when the *w*-value equals one and zero, respectively.

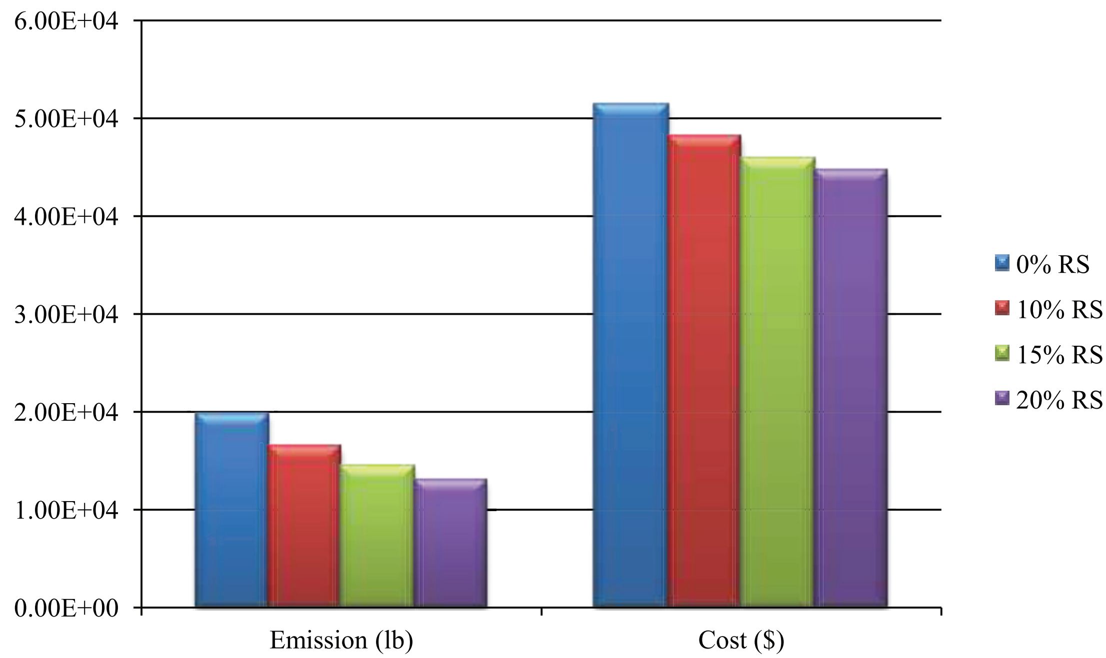


Figure 5. 23: Operating fuel's cost and emission reductions due to different RS contribution's levels obtained by the ABC* algorithm when the *w*-value equals 0.50.

5.5 SUMMARY

In this chapter, a novel constrained search-tactic was proposed to solve the DED and DEED problems. Utilizing the offered search-tactic into two recently introduced meta-heuristic algorithms was presented as well. The suggested search-tactic changed the objective function from minimizing the considered aim to minimizing the system's equality constraints' violations. This substitution was utilized temporally to avoid further exploitation of infeasible solutions or local solutions entrapments.

The ABC and SDOA algorithms were designated for evaluating the effectiveness of the suggested constrained search-tactic. The six-unit and 15-unit systems were adopted to reveal that efficiency. In both systems, utilizing the proposed constrained search-tactic via both algorithms successfully attained the optimal solutions with additional reduction in the required CPU time and considered objective function. The SDOA* (with the integration of the offered search-tactic) obtained slightly enhanced results in terms of economic operating output power dispatched for both systems with respect to those attained using other well-known techniques. From the promising results in this chapter as well as those of chapter four, the SDOA algorithm has a potential to be applied to complex and high dimensional optimization problems in future research.

The DEED problem has been solved by the ABC algorithm with the assistance of the offered constrained search-tactic. In this thesis, the considered emission was based on NO_x effect. However, fossil fueled plants have been held accountable for discharging other harmful gases, e.g., CO_2 and SO_x. These pollutant gases' emissions will be used in future research, as elucidated in the next chapter. The results obtained by the proposed algorithm showed a practical meaning of achieving the desired objective function with least power mismatch values. Comparative analyses confirmed that the results obtained by the suggested techniques outperformed those of other well-known methods. Several test cases were used to observe the impact of integrating different RS (wind power farms) applications. In all cases tested, significant reductions in the operating fuel's cost and emission as well as total system's losses were achieved once the contribution of the RS practices was taken into account. As mentioned earlier, the competing objective functions have more than one solution. The "exact" optimal solution is highly depended on the decision-makers' preferences. These preferences, however, are significantly influenced by political, social, and economic factors.

CHAPTER 6: CONCLUSIONS AND RECOMMENDATIONS

6.1 CONCLUSIONS AND CONTRIBUTIONS

Initially, the operation of electric power industries faced new and challenging policies. Environmental awareness, for example, dramatically altered these industries' ultimate objective. Obtaining a break-even point between economic goals and emission control has been an ongoing concern. However, the optimal strategy followed by one system may not be applicable or sufficient in others because of political, social, and economic factors. A precise model that accounts for practical constraints and avoids oversimplified assumptions has always been essential. The dynamic economic and emission dispatch problems represented the practical meaning of optimal operation and control of on-line generation units to meet the demand of power system networks. The prevalence of nonlinear and non-convex characteristics that drove the DED problem's search-space encountered many local extrema solutions resulted in a challenging task to determine a global solution.

Deterministic optimization algorithms have been utilized in solving the dynamic economic and emission dispatch problems at early stages. Then, unconventional (heuristic and/or meta-heuristic) optimization techniques shared that utilization. That utilization shift was due to the merits of unconventional methods such as independent of initial solutions, derivative-free, and easy to implement. After that, a combination of two or more optimization tools dominated the trend in solving the problem. The complexity of the DED problem was one reason of such hybrid methods' dominance. Another cause was to enhance the solution quality by overcoming the limitations of each of the individual techniques.

Natural phenomena have been the main inspiration of most sufficient meta-heuristic optimization techniques. The ABC algorithm was inspired by the intelligent foraging honeybee swarms. Although the ABC algorithm is considered a recently introduced meta-heuristic optimization tool, it has been adopted in a variety of real-world optimization applications, and is growing rapidly as evident in the literature. However, the ABC method has not been addressed well in electric power system optimization areas. This thesis attempted to fill this gap. The ABC algorithm has only two parameters to be adjusted. Therefore, updating them towards the most effective values has a higher likelihood of success than other competing meta-heuristic algorithms.

A new naturally inspired meta-heuristic algorithm has been proposed in this thesis. The SDOA method is motivated by the intelligent behaviour or survival of sensory-deprived human beings. A set of benchmark optimization functions was examined to confirm the efficiency of the suggested algorithm. In addition, the results obtained by the SDOA algorithm outperformed or matched those attained using other well-known meta-heuristic optimization methods.

This thesis started by offering the state-of-the-art survey of both the dynamic economic and emission dispatch problems and the ABC algorithm applications. Then, a novel meta-heuristic (SDOA) optimization algorithm was proposed. A new constrained search-tactic was also presented to enhance the utilized algorithms' performances. Various experimental tests and comparisons with other well-known tools validated the efficiency of the proposed methods in solving complex, nonlinear, and non-convex optimization problems. This thesis has resulted in several contributions such as:

- Offered an overview of the state-of-the-art overview of the DED problem. Various techniques were used, and the reviewed literature was classified into three categories: deterministic, heuristic, and hybrid methods. Advantages, disadvantages, and considered constraints of each were also highlighted.
- Studied, analyzed, and implemented the ABC algorithm in solving different types of electric power system optimization problems. It was used to solve a mixed integer nonlinear optimization problem such as the optimal allocation of DG application. In addition, it utilized the ABC algorithm in solving the DED and DEED problems, by considering the single and multi-objective functions respectively. A comprehensive survey of the literature that employed the ABC algorithm and categorization of these areas of application was another significant contribution of this thesis.

- Proposed a novel (SDOA) meta-heuristic optimization algorithm. A set of benchmark optimization functions was examined to confirm the efficacy of the suggested algorithm. In addition, the results obtained by the proposed algorithm were compared with those obtained using other well-known meta-heuristic optimization algorithms. It implemented the proposed SDOA in solving both the mixed integer nonlinear optimization problem (i.e., DG allocation) and the DED problem, and analyzed the potential of the SDOA algorithm in high dimensional applications.
- Integrated a renewable source in solving the DED and DEED problems while emphasizing the impact of such integration on both objectives. Moreover, a new constraint-handling strategy in solving the DED and DEED problems was proposed. Evaluation of the effectiveness of this strategy, via two different meta-heuristic optimization algorithms, was also conducted.

6.2 Recommendations for Future Research

The contributions of this thesis may be extended for future research in the following areas:

– The ABC and SDOA algorithm can be utilized to solve other electric power system optimization areas such as UC and OPF. Therefore, the SDOA, specifically, can be further analyzed and its potential highlighted. Integrating these algorithms in ANN and fuzzy-based applications would be another direction for future research.

– The DED problem can be subjected to additional constraints or objective functions. In other words, investigating the effect and cost of integrating renewable sources as a system's spinning reserve would be an intriguing research subject. The integration of renewable sources can be based on a probability-based technique, such as a chance-constrained method.

– The DEED can be solved as a single objective function, with the emission's amount treated as inequality constraints. By including various gas emissions, e.g., CO_2 and SO_x in the considered problem, it can be formulated as multiple objective functions or inequality constraints.

- The dynamic economic and emission problem can be extended to consider the load demand forecast, UC, electricity market, and automatic generation control (AGC) applications. Furthermore, the transmission losses of the system can be expressed using the OPF formulation.
- The constraints of the DED problem can be handled adaptively, i.e., dynamically. In addition, including the feasibility-based criteria would be interesting in such a complex, nonlinear, and non-convex optimization problem.
- The combination of the ABC or SDOA algorithms with another deterministic tool can be done in the following assumptions. First, either ABC or SDOA would be used as the main search mechanism, and the deterministic method would handle the local search task. Second, the ABC and SDOA can be hybridized to improve each algorithm's performance. Third, the ABC and/or SDOA methods can be divided into parallel search engines using multiple processors.
- The contributions of renewable sources can be based on a designated time interval(s) with a high likelihood of certainty. In addition, another dimension of future research is to consider different types of renewable sources' applications.

BIBLIOGRAPHY

[1] T.E. Bechert and H.G. Kwatny, "On the Optimal Dynamic Dispatch of Real Power," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-91, no. 3, pp. 889-898, 1972.

[2] T.E. Bechert and Nanming Chen, "Area Automatic Generation Control by Multi-Pass Dynamic Programming," *Power Apparatus and Systems, IEEE Transactions on,* vol. 96, no. 5, pp. 1460-1469, 1977.

[3] N.P. Padhy, "Unit Commitment-a Bibliographical Survey," *Power Systems, IEEE Transactions on,* vol. 19, no. 2, pp. 1196-1205, 2004.

[4] J.A. Momoh, R. Adapa and M.E. El-Hawary, "A Review of Selected Optimal Power Flow Literature to 1993. I. Nonlinear and Quadratic Programming Approaches," *Power Systems, IEEE Transactions on,* vol. 14, no. 1, pp. 96-104, 1999.

[5] J.A. Momoh, M.E. El-Hawary and R. Adapa, "A Review of Selected Optimal Power Flow Literature to 1993. II. Newton, Linear Programming and Interior Point Methods," *Power Systems, IEEE Transactions on,* vol. 14, no. 1, pp. 105-111, 1999.

[6] M.R. AlRashidi and M.E. El-Hawary, "Applications of Computational Intelligence Techniques for Solving the Revived Optimal Power Flow Problem," *Electr. Power Syst. Res.,* vol. 79, no. 4, pp. 694-702, 2009.

[7] A. Mahor, V. Prasad and S. Rangnekar, "Economic Dispatch using Particle Swarm Optimization: A Review," *Renewable and Sustainable Energy Reviews,* vol. 13, no. 8, pp. 2134-2141, 2009.

[8] H. Altun and T. Yalcinoz, "Implementing Soft Computing Techniques to Solve Economic Dispatch Problem in Power Systems," *Expert Syst. Appl.,* vol. 35, no. 4, pp. 1668-1678, 2008.

[9] X. Xia and A.M. Elaiw, "Optimal Dynamic Economic Dispatch of Generation: A Review," *Electr. Power Syst. Res.,* vol. In Press, Corrected Proof, 2010.

[10] I.A. Farhat and M.E. El-Hawary, "Optimization Methods Applied for Solving the Short-Term Hydrothermal Coordination Problem," *Electr. Power Syst. Res.,* vol. 79, no. 9, pp. 1308-1320, 2009.

[11] P.P.J. van den Bosch, "Optimal Dynamic Dispatch Owing to Spinning-Reserve and Power-Rate Limits," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-104, no. 12, pp. 3395-3401, 1985.

[12] D.W. Ross and Sungkook Kim, "Dynamic Economic Dispatch of Generation," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-99, no. 6, pp. 2060-2068, 1980.

[13] W.G. Wood, "Spinning Reserve Constrained Static and Dynamic Economic Dispatch," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-101, no. 2, pp. 381-388, 1982.

[14] F. Li, R. Morgan and D. Williams, "Hybrid Genetic Approaches to Ramping Rate Constrained Dynamic Economic Dispatch," *Electr. Power Syst. Res.,* vol. 43, no. 2, pp. 97-103, 1997.

[15] Y.H. Song and I. Yu, "Dynamic Load Dispatch with Voltage Security and Environmental Constraints," *Electr. Power Syst. Res.,* vol. 43, no. 1, pp. 53-60, 1997.

[16] R.R. Shoults, R.K. Chakravarty and R. Lowther, "Quasi-Static Economic Dispatch using Dynamic Programming with an Improved Zoom Feature," *Electr. Power Syst. Res.,* vol. 39, no. 3, pp. 215-222, 1996.

[17] W.R. Barcelo and P. Rastgoufard, "Dynamic Economic Dispatch using the Extended Security Constrained Economic Dispatch Algorithm," *Power Systems, IEEE Transactions on,* vol. 12, no. 2, pp. 961-967, 1997.

[18] W.R. Barcelo and P. Rastgoufard, "Control Area Performance Improvement by Extended Security Constrained Economic Dispatch," *Power Systems, IEEE Transactions on,* vol. 12, no. 1, pp. 120-128, 1997.

[19] R.W. Ferrero and S.M. Shahidehpour, "Dynamic Economic Dispatch in Deregulated Systems," *International Journal of Electrical Power & Energy Systems,* vol. 19, no. 7, pp. 433-439, 1997.

[20] K. Chandram, N. Subrahmanyam and M. Sydulu, "Dynamic Economic Dispatch by Equal Embedded Algorithm," *Electrical and Computer Engineering, 2006. ICECE '06. International Conference on,* pp. 21-24, 2006.

[21] R.P. Brent, Algorithms for Minimization without Derivatives, N.J.: Prentice-Hall, Englewood Cliffs, 1973.

[22] K. Chandram, N. Subrahmanyam and M. Sydulu, "Brent Method for Dynamic Economic Dispatch with Transmission Losses," *Transmission and Distribution Conference and Exposition, 2008. T&D. IEEE/PES,* pp. 1-5, 2008.

[23] Xiaohua Xia, Jiangfeng Zhang and A. Elaiw, "A Model Predictive Control Approach to Dynamic Economic Dispatch Problem," *PowerTech, 2009 IEEE Bucharest,* pp. 1-7, 2009.

[24] X. Xia, J. Zhang and A. Elaiw, "An Application of Model Predictive Control to the Dynamic Economic Dispatch of Power Generation," *Control Eng. Pract.,* vol. 19, no. 6, pp. 638-648, 2011.

[25] S. Hemamalini and S.P. Simon, "Dynamic Economic Dispatch using Maclaurin Series Based Lagrangian Method," *Energy Conversion and Management,* vol. 51, no. 11, pp. 2212-2219, 2010.

[26] E. Mezura-Montes, Constraint-Handling in Evolutionary Optimization, Heidelberg: Springer Berlin, 2009.

[27] K. Deb, "An Efficient Constraint Handling Method for Genetic Algorithms," *Comput. Methods Appl. Mech. Eng.,* vol. 186, no. 2-4, pp. 311-338, 2000.

[28] W. Ongsakul and J. Tippayachai, "Parallel Micro Genetic Algorithm Based on Merit Order Loading Solutions for Constrained Dynamic Economic Dispatch," *Electr. Power Syst. Res.,* vol. 61, no. 2, pp. 77-88, 2002.

[29] X. Yuan, L. Wang, Y. Zhang and Y. Yuan, "A Hybrid Differential Evolution Method for Dynamic Economic Dispatch with Valve-Point Effects," *Expert Syst. Appl.,* vol. 36, no. 2, Part 2, pp. 4042-4048, 2009.

[30] Zwe-Lee Gaing, "Constrained Dynamic Economic Dispatch Solution using Particle Swarm Optimization," *Power Engineering Society General Meeting, 2004. IEEE,* pp. 153-158 Vol.1, 2004.

[31] M. Basu, "Particle Swarm Optimization Based Goal-Attainment Method for Dynamic Economic Emission Dispatch," *Electric Power Components and Systems,* vol. 34, no. 9, pp. 1015-1025, 2006.

[32] V. Chankong and Y. Haimes, Multiobjective Decision Making: Theory and Methodology, New York: North-Holland, 1983.

[33] M. Basu, "Dynamic Economic Emission Dispatch using Nondominated Sorting Genetic Algorithm-II," *International Journal of Electrical Power & Energy Systems,* vol. 30, no. 2, pp. 140-149, 2008.

[34] Bo Zhao, Chuangxin Guo and Yijia Cao, "Dynamic Economic Dispatch in Electricity Market using Particle Swarm Optimization Algorithm," *Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on,* vol. 6, pp. 5050-5054 Vol.6, 2004.

[35] F. Li, R. Morgan and D. Williams, "Towards More Cost Saving Under Stricter Ramping Rate Constraints of Dynamic Economic Dispatch Problems-a Genetic Based Approach," *Genetic Algorithms in Engineering Systems: Innovations and Applications, 1997. GALESIA 97. Second International Conference on (Conf. Publ. no. 446),* pp. 221-225, 1997.

[36] K. Shailti Swamp and A. Natarajan, "Constrained Optimization using Evolutionary Programming for Dynamic Economic Dispatch," *Intelligent Sensing and Information Processing, 2005. Proceedings of 2005 International Conference on,* pp. 314-319, 2005.

[37] A.M.A.A. Joned, I. Musirin and Titik Khawa Abdul Rahman, "Solving Dynamic Economic Dispatch using Evolutionary Programming," *Power and Energy Conference, 2006. PECon '06. IEEE International,* pp. 144-149, 2006.

[38] C.K. Panigrahi, P.K. Chattopadhyay, R.N. Chakrabarti and M. Basu, "Simulated Annealing Technique for Dynamic Economic Dispatch," *Electric Power Components and Systems,* vol. 34, no. 5, pp. 577-587, 2006.

[39] G.S.S. Babu, D.B. Das and C. Patvardhan, "Dynamic Economic Dispatch Solution using an Enhanced Real-Quantum Evolutionary Algorithm," *Power System Technology and IEEE Power India Conference, 2008. POWERCON 2008. Joint International Conference on,* pp. 1-6, 2008.

[40] S. Pothiya, I. Ngamroo and W. Kongprawechnon, "Application of Multiple Tabu Search Algorithm to Solve Dynamic Economic Dispatch Considering Generator Constraints," *Energy Conversion and Management,* vol. 49, no. 4, pp. 506-516, 2008.

[41] Xiuqin Shang, Jiangang Lu and Youxian Sun, "A Preference-Based Non-Dominated Sorting Genetic Algorithm on Dynamic Economic Dispatch," *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on,* pp. 2794-2797, 2008.

[42] X. Yuan, L. Wang, Y. Yuan, Y. Zhang, B. Cao and B. Yang, "A Modified Differential Evolution Approach for Dynamic Economic Dispatch with Valve-Point Effects," *Energy Conversion and Management,* vol. 49, no. 12, pp. 3447-3453, 2008.

[43] J.-. Chiou, "A Variable Scaling Hybrid Differential Evolution for Solving Large-Scale Power Dispatch Problems," *Generation, Transmission & Distribution, IET,* vol. 3, no. 2, pp. 154-163, 2009.

[44] J.S. Alsumait, M. Qasem, J.K. Sykulski and A.K. Al-Othman, "An Improved Pattern Search Based Algorithm to Solve the Dynamic Economic Dispatch Problem with Valve-Point Effect," *Energy Conversion and Management,* vol. 51, no. 10, pp. 2062-2067, 2010.

[45] M. Basu, "Artificial Immune System for Dynamic Economic Dispatch," *International Journal of Electrical Power & Energy Systems,* vol. 33, no. 1, pp. 131-136, 2011.

[46] S. Hemamalini and S.P. Simon, "Dynamic Economic Dispatch using Artificial Immune System for Units with Valve-Point Effect," *International Journal of Electrical Power & Energy Systems,* vol. In Press, Corrected Proof, pp. 1-7, 2011.

[47] F. Kappel and A.V. Kuntsevich, "An Implementation of Shor's r-Algorithm," *Computational Optimization and Applications,* vol. 15, no. 2, pp. 193-205, 2000.

[48] T.A.A. Victoire and A.E. Jeyakumar, "Reserve Constrained Dynamic Dispatch of Units with Valve-Point Effects," *Power Systems, IEEE Transactions on,* vol. 20, no. 3, pp. 1273-1282, 2005.

[49] T.A.A. Victoire and A.E. Jeyakumar, "A Modified Hybrid EP–SQP Approach for Dynamic Dispatch with Valve-Point Effect," *International Journal of Electrical Power & Energy Systems,* vol. 27, no. 8, pp. 594-601, 2005.

[50] T.A.A. Victoire and A.E. Jeyakumar, "Deterministically Guided PSO for Dynamic Dispatch Considering Valve-Point Effect," *Electr. Power Syst. Res.,* vol. 73, no. 3, pp. 313-322, 2005.

[51] M. Basu, "Dynamic Economic Emission Dispatch using Evolutionary Programming and Fuzzy Satisfying Method," *International Journal of Emerging Electric Power Systems,* vol. 8, no. 4, pp. 1-15, 2007.

[52] W. Lin and S. Chen, "Bid-Based Dynamic Economic Dispatch with an Efficient Interior Point Algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 24, no. 1, pp. 51-57, 2002.

[53] C.B. Somuah and N. Khunaizi, "Application of Linear Programming Redispatch Technique to Dynamic Generation Allocation," *Power Systems, IEEE Transactions on,* vol. 5, no. 1, pp. 20-26, 1990.

[54] Ruey-Hsum Liang, "A Neural-Based Redispatch Approach to Dynamic Generation Allocation," *Power Systems, IEEE Transactions on,* vol. 14, no. 4, pp. 1388-1393, 1999.

[55] F. Li and R.K. Aggarwal, "Fast and Accurate Power Dispatch using a Relaxed Genetic Algorithm and a Local Gradient Technique," *Expert Syst. Appl.,* vol. 19, no. 3, pp. 159-165, 2000.

[56] X.S. Han, H.B. Gooi and D.S. Kirschen, "Dynamic Economic Dispatch: Feasible and Optimal Solutions," *Power Engineering Society Summer Meeting, 2001. IEEE,* vol. 3, pp. 1704 vol.3, 2001.

[57] W. Ongsakul and N. Ruangpayoongsak, "Constrained Dynamic Economic Dispatch by Simulated annealing/genetic Algorithms," *Power Industry Computer Applications, 2001. PICA 2001. Innovative Computing for Power - Electric Energy Meets the Market. 22nd IEEE Power Engineering Society International Conference on,* pp. 207-212, 2001.

[58] P. Attaviriyanupap, H. Kita, E. Tanaka and J. Hasegawa, "A Hybrid EP and SQP for Dynamic Economic Dispatch with Nonsmooth Fuel Cost Function," *Power Systems, IEEE Transactions on,* vol. 17, no. 2, pp. 411-416, 2002.

[59] P. Attaviriyanupap, H. Kita, E. Tanaka and J. Hasegawa, "A Fuzzy-Optimization Approach to Dynamic Economic Dispatch Considering Uncertainties," *Power Systems, IEEE Transactions on,* vol. 19, no. 3, pp. 1299-1307, 2004.

[60] Guoli Zhang, Hai Yan Lu, Gengyin Li and Hong Xie, "A New Hybrid Real-Coded Genetic Algorithm and Application in Dynamic Economic Dispatch," *Intelligent Control and Automation, 2006. WCICA 2006. the Sixth World Congress on,* vol. 1, pp. 3627-3632, 2006.

[61] C.-. Chen, "Simulated Annealing-Based Optimal Wind-Thermal Coordination Scheduling," *Generation, Transmission & Distribution, IET,* vol. 1, no. 3, pp. 447-455, 2007.

[62] A.Y. Abdelaziz, M.Z. Kamh, S.F. Mekhamer and M.A.L. Badr, "A Hybrid HNN-QP Approach for Dynamic Economic Dispatch Problem," *Electr. Power Syst. Res.,* vol. 78, no. 10, pp. 1784-1788, 2008.

[63] A.Y. Abdelaziz, S.F. Mekhamer, M.Z. Kamh and M.A.L. Badr, "A Hybrid Hopfield Neural Network-Quadratic Programming Approach for Dynamic Economic Dispatch Problem," *Power System Conference, 2008. MEPCON 2008. 12th International Middle-East,* pp. 565-570, 2008.

[64] Zwe-Lee Gaing and Ting-Chia Ou, "Dynamic Economic Dispatch Solution using Fast Evolutionary Programming with Swarm Direction," *Industrial Electronics and Applications, 2009. ICIEA 2009. 4th IEEE Conference on,* pp. 1538-1544, 2009.

[65] Y. Lu, J. Zhou, H. Qin, Y. Li and Y. Zhang, "An Adaptive Hybrid Differential Evolution Algorithm for Dynamic Economic Dispatch with Valve-Point Effects," *Expert Syst. Appl.,* vol. In Press, Accepted Manuscript, 2009.

[66] A.I. Selvakumar and K. Thanushkodi, "A New Particle Swarm Optimization Solution to Nonconvex Economic Dispatch Problems," *Power Systems, IEEE Transactions on,* vol. 22, no. 1, pp. 42-51, 2007.

[67] C.A. Coello Coello, "Theoretical and Numerical Constraint-Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art," *Comput. Methods Appl. Mech. Eng.,* vol. 191, no. 11-12, pp. 1245-1287, 2002.

[68] S. Titus and A.E. Jeyakumar, "A Hybrid EP-PSO-SQP Algorithm for Dynamic Dispatch Considering Prohibited Operating Zones," *Electric Power Components and Systems,* vol. 36, no. 5, pp. 449-467, 2008.

[69] S. Sivasubramani and K.S. Swarup, "Hybrid SOA–SQP Algorithm for Dynamic Economic Dispatch with Valve-Point Effects," *Energy,* vol. 35, no. 12, pp. 5031-5036, 2010.

[70] C. Dai, Y. Zhu and W. Chen, "Seeker Optimization Algorithm," *Lect. Notes Comput. Sci.,* vol. 4456, pp. 167-176, 2007.

[71] P. Praveena, K. Vaisakh and S.R.M. Rao, "A Bacterial Foraging and PSO-DE Algorithm for Solving Dynamic Economic Dispatch Problem with Valve-Point Effects," *in Integrated Intelligent Computing (ICIIC), 2010 First International Conference on,* pp. 227-232, 2010.

[72] K.M. Passino, "Biomimicry of Bacterial Foraging for Distributed Optimization and Control," *Control Systems, IEEE,* vol. 22, no. 3, pp. 52-67, 2002.

[73] P. Praveena, K. Vaisakh and S. Rama Mohana Rao, "A Bacterial Foraging PSO-DE Algorithm for Solving Dynamic Economic Dispatch Problem with Security Constraints," *in Power Electronics, Drives and Energy Systems (PEDES) & 2010 Power India, 2010 Joint International Conference on,* pp. 1-7, 2010.

[74] V. Ravikumar Pandi and B.K. Panigrahi, "Dynamic Economic Load Dispatch using Hybrid Swarm Intelligence Based Harmony Search Algorithm," *Expert Syst. Appl.,* vol. In Press, Corrected Proof, pp. 1-6, 2011.

[75] Z.W. Geem, Music-inspired harmony search algorithm: Theory and pplications, Berlin: Springer, 2009.

[76] M.G.H. Omran and M. Mahdavi, "Global-Best Harmony Search," *Applied Mathematics and Computation,* vol. 198, no. 2, pp. 643-656, 2008.

[77] Y. Lu, J. Zhou, H. Qin, Y. Wang and Y. Zhang, "Chaotic Differential Evolution Methods for Dynamic Economic Dispatch with Valve-Point Effects," *Eng Appl Artif Intell,* vol. 24, no. 2, pp. 378-387, 2011.

[78] M.S. Tavazoei and M. Haeri, "Comparison of Different One-Dimensional Maps as Chaotic Search Pattern in Chaos Optimization Algorithms," *Applied Mathematics and Computation,* vol. 187, no. 2, pp. 1076-1085, 2007.

[79] J.H. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, 1975.

[80] M. Dorigo, "Optimization, Learning and Natural Algorithms," *Ph. D. Thesis,* 1992.

[81] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *in Neural Networks, 1995. Proceedings., IEEE International Conference on,* pp. 1942-1948 vol.4, 1995.

[82] D. Karaboga, "An Idea based on Honey Bee Swarm for Numerical Optimization," Erciyes University, Engineering Faculty, Computer Engineering Department., Tech. Rep. TR06, pp. 1-10, 2005.

[83] X.S. Yang, Nature-Inspired Metaheuristic Algorithms, United Kingdom: Luniver Press, 2008.

[84] D. Karaboga and B. Basturk, "A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm," *J. Global Optimiz.,* vol. 39, no. 3, pp. 459-471, 2007.

[85] D. Karaboga and B. Basturk, "Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems," *Foundations of Fuzzy Logic and Soft Computing,* pp. 789-798, 2007.

[86] I.C. Trelea, "The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection," *Information Processing Letters,* vol. 85, no. 6, pp. 317-325, 2003.

[87] N. Karaboga, "A New Design Method Based on Artificial Bee Colony Algorithm for Digital IIR Filters," *Journal of the Franklin Institute,* vol. 346, no. 4, pp. 328-348, 2009.

[88] D. Karaboga and B. Basturk, "On the Performance of Artificial Bee Colony (ABC) Algorithm," *Applied Soft Computing,* vol. 8, no. 1, pp. 687-697, 2008.

[89] D. Karaboga and B. Akay, "A Comparative Study of Artificial Bee Colony Algorithm," *Applied Mathematics and Computation,* vol. 214, no. 1, pp. 108-132, 2009.

[90] K.V. Frisch, The Dancing Bees: An Account of the Life and Senses of Honey Bee, Harcourt, Brace, 1953.

[91] M. Lindauer, Communication among Social Bees, Harvard University Press, 1961.

[92] K.V. Frisch, Bees Their Vision, Chemical Senses, and Language, United Kingdom: Cornell University Press, 1971.

[93] K.V. Frisch, The Dance Language and Orientation of Bees, Harvard University Press, 1976.

[94] T.D. Seeley, Honeybee Ecology: A study of Adaptation in Social Life, Princeton: Princeton University Press, 1985.

[95] Y. Yonezawa and T. Kikuchi, "Ecological Algorithm for Optimal Ordering used by Collective Honey Bee Behavior," *in Micro Machine and Human Science, 1996., Proceedings of the Seventh International Symposium,* pp. 249-256, 1996.

[96] T. Sato and M. Hagiwara, "Bee System: Finding Solution by a Concentrated Search," *in Systems, Man, and Cybernetics, 1997. 'Computational Cybernetics and Simulation'., 1997 IEEE International Conference on,* pp. 3954-3959 vol.4, 1997.

[97] D.J.T. Sumpter and D.S. Broomhead, "Formalising the Link between Worker and Society in Honey Bee Colonies," *Multi-Agent Systems and Agent-Based Simulation,* vol. 1534, pp. 95-110, 1998.

[98] A. Walker, J. Hallam and D. Willshaw, "Bee-Havior in a Mobile Robot: The Construction of a Self-Organized Cognitive Map and its use in Robot Navigation within a Complex, Natural Environment," *in Neural Networks, 1993., IEEE International Conference on,* pp. 1451-1456 vol.3, 1993.

[99] D. Karaboga and B. Akay, "A Survey: Algorithms Simulating Bee Swarm Intelligence," *Artif. Intell. Rev.,* vol. 31, no. 1, pp. 61-85, 2009.

[100] D. Karaboga and B. Akay, "A Modified Artificial Bee Colony (ABC) Algorithm for Constrained Optimization Problems," *Applied Soft Computing,* vol. 11, no. 3, pp. 3021-3031, 2011.

[101] B. Akay and D. Karaboga, "Artificial Bee Colony Algorithm for Large-Scale Problems and Engineering Design Optimization," *J. Intell. Manuf.,* 2010.

[102] J.A. Pacurib, G.M.M. Seno and J.P.T. Yusiong, "Solving Sudoku Puzzles using Improved Artificial Bee Colony Algorithm," *in Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on,* pp. 885-888, 2009.

[103] H.A.A. Bahamish, R. Abdullah and R.A. Salam, "Protein Tertiary Structure Prediction using Artificial Bee Colony Algorithm," *in Modelling & Simulation, 2009. AMS '09. Third Asia International Conference on,* pp. 258-263, 2009.

[104] H.A.A. Bahamish and R. Abdullah, "Prediction of C-Peptide Structure using Artificial Bee Colony Algorithm," *in Information Technology (ITSim), 2010 International Symposium in,* pp. 754-759, 2010.

[105] Yun-Ling Liu and Lan Tao, "An Improved Parallel Simulated Annealing Algorithm used for Protein Structure Prediction," *in Machine Learning and Cybernetics, 2006 International Conference on,* pp. 2335-2338, 2006.

[106] B. Akay and D. Karaboga, "Solving Integer Programming Problems by using Artificial Bee Colony Algorithm," *AI*IA 2009: Emergent Perspectives in Artificial Intelligence,* pp. 355-364, 2009.

[107] B. Akay and D. Karaboga, "Parameter Tuning for the Artificial Bee Colony Algorithm," *Computational Collective Intelligence.Semantic Web, Social Networks and Multiagent Systems,* pp. 608-619, 2009.

[108] D. Karaboga, B. Akay and C. Ozturk, "Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks," *Modeling Decisions for Artificial Intelligence,* pp. 318-329, 2007.

[109] B. Akay and D. Karaboga, "A Modified Artificial Bee Colony Algorithm for Real-Parameter Optimization," *Inf. Sci.,* vol. In Press, Corrected Proof, 2010.

[110] Haiyan Quan and Xinling Shi, "On the Analysis of Performance of the Improved Artificial-Bee-Colony Algorithm," *in Natural Computation, 2008. ICNC '08. Fourth International Conference on,* pp. 654-658, 2008.

[111] B. Alatas, "Chaotic Bee Colony Algorithms for Global Numerical Optimization," *Expert Syst. Appl.,* vol. In Press, Corrected Proof, 2010.

[112] G. Zhu and S. Kwong, "Gbest-Guided Artificial Bee Colony Algorithm for Numerical Function Optimization," *Applied Mathematics and Computation,* vol. 217, no. 7, pp. 3166-3173, 2010.

[113] Wenping Zou, Yunlong Zhu, Hanning Chen and Zhu Zhu, "Cooperative Approaches to Artificial Bee Colony Algorithm," *in Computer Application and System Modeling (ICCASM), 2010 International Conference on,* pp. V9-44-V9-48, 2010.

[114] M. El-Abd, "A Cooperative Approach to the Artificial Bee Colony Algorithm," *in Evolutionary Computation (CEC), 2010 IEEE Congress on,* pp. 1-5, 2010.

[115] M.S. Alam, M.W. Ul Kabir and M.M. Islam, "Self-Adaptation of Mutation Step Size in Artificial Bee Colony Algorithm for Continuous Function Optimization," *in Computer and Information Technology (ICCIT), 2010 13th International Conference on,* pp. 69-74, 2010.

[116] E. Mezura-Montes and R.E. Velez-Koeppel, "Elitist Artificial Bee Colony for Constrained Real-Parameter Optimization," *in Evolutionary Computation (CEC), 2010 IEEE Congress on,* pp. 1-8, 2010.

[117] A. Banharnsakun, T. Achalakul and B. Sirinaovakul, "The Best-so-Far Selection in Artificial Bee Colony Algorithm," *Applied Soft Computing,* vol. 11, no. 2, pp. 2888-2901, 2011.

[118] Xiaojun Bi and Yanjiao Wang, "An Improved Artificial Bee Colony Algorithm," *in Computer Research and Development (ICCRD), 2011 3rd International Conference on,* pp. 174-177, 2011.

[119] S.K. Nayak, K.R. Krishnanand, B.K. Panigrahi and P.K. Rout, "Application of Artificial Bee Colony to Economic Load Dispatch Problem with Ramp Rate Limits and Prohibited Operating Zones," *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on,* pp. 1237-1242, 2009.

[120] F. S. Abu-Mouti and M.E. El-Hawary, "Modified Artificial Bee Colony Algorithm for Optimal Distributed Generation Sizing and Allocation in Distribution Systems," *in Electrical Power & Energy Conference (EPEC), 2009 IEEE,* pp. 1-9, 2009.

[121] F. S. Abu-Mouti and M.E. El-Hawary, "Optimal Distributed Generation Allocation and Sizing in Distribution Systems via Artificial Bee Colony Algorithm," *Power Delivery, IEEE Transactions on,* vol. 26, no. 4, pp. 2090-2101, 2011.

[122] F. S. Abu-Mouti and M.E. El-Hawary, "A Priority-Ordered Constrained Search Technique for Optimal Distributed Generation Allocation in Radial Distribution Feeder Systems," *in IEEE 23rd Canadian Conference on Electrical and Computer Engineering,* pp. 1-7, 02-05 May 2010.

[123] S.B. Rao, S.V.L. Narasimham and M. Ramalingaraju, "Optimization of Distribution Network Configuration for Loss Reduction using Artificial Bee Colony Algorithm," *International Journal of Electrical Power and Energy Systems Engineering,* vol. 1, no. 2, pp. 116-122, 2008.

[124] Nguyen Tung Linh and Nguyen Quynh Anh, "Application Artificial Bee Colony Algorithm (ABC) for Reconfiguring Distribution Network," *in Computer Modeling and Simulation, 2010. ICCMS '10. Second International Conference on,* pp. 102-106, 2010.

[125] C. Sumpavakup, I. Srikun and S. Chusanapiputt, "A Solution to the Optimal Power Flow using Artificial Bee Colony Algorithm," *in Power System Technology (POWERCON), 2010 International Conference on,* pp. 1-5, 2010.

[126] H. Narasimhan, "Parallel Artificial Bee Colony (PABC) Algorithm," *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on,* pp. 306-311, 2009.

[127] A. Banharnsakun, T. Achalakul and B. Sirinaovakul, "Artificial Bee Colony Algorithm on Distributed Environments," *in Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress on,* pp. 13-18, 2010.

[128] Ruhai Luo, Tien-Szu Pan, Pei-Wei Tsai and Jeng-Shyang Pan, "Parallelized Artificial Bee Colony with Ripple-Communication Strategy," *in Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference on,* pp. 350-353, 2010.

[129] C. Xu and H. Duan, "Artificial Bee Colony (ABC) Optimized Edge Potential Function (EPF) Approach to Target Recognition for Low-Altitude Aircraft," *Pattern Recog. Lett.,* vol. In Press, Corrected Proof, 2009.

[130] C. Chidambaram and H.S. Lopes, "A New Approach for Template Matching in Digital Images using an Artificial Bee Colony Algorithm," *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on,* pp. 146-151, 2009.

[131] C. Zhang, D. Ouyang and J. Ning, "An Artificial Bee Colony Approach for Clustering," *Expert Syst. Appl.,* vol. 37, no. 7, pp. 4761-4767, 2010.

[132] D. Karaboga and C. Ozturk, "A Novel Clustering Approach: Artificial Bee Colony (ABC) Algorithm," *Applied Soft Computing,* vol. In Press, Corrected Proof, 2009.

[133] Xiujuan Lei, Xu Huang and Aidong Zhang, "Improved Artificial Bee Colony Algorithm and its Application in Data Clustering," *in Bio-Inspired Computing: Theories and Applications (BIC-TA), 2010 IEEE Fifth International Conference on,* pp. 514-521, 2010.

[134] A. Singh, "An Artificial Bee Colony Algorithm for the Leaf-Constrained Minimum Spanning Tree Problem," *Applied Soft Computing,* vol. 9, no. 2, pp. 625-631, 2009.

[135] B.A. Julstrom, "Codings and Operators in Two Genetic Algorithms for the Leaf Constrained Minimum Spanning Tree Problem," *International Journal of Applied Mathematics and Computer Science,* vol. 14, no. 3, pp. 385-396, 2004.

[136] T. Hsieh, H. Hsiao and W. Yeh, "Forecasting Stock Markets using Wavelet Transforms and Recurrent Neural Networks: An Integrated System Based on Artificial Bee Colony Algorithm," *Applied Soft Computing,* vol. 11, no. 2, pp. 2510-2525, 2011.

[137] V.J. Manoj and E. Elias, "Artificial Bee Colony Algorithm for the Design of Multiplier-Less Nonuniform Filter Bank Transmultiplexer," *Inf. Sci.,* vol. In Press, Corrected Proof, 2011.

[138] S.K. Udgata, S.L. Sabat and S. Mini, "Sensor Deployment in Irregular Terrain using Artificial Bee Colony Algorithm," *in Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on,* pp. 1309-1314, 2009.

[139] S.L. Sabat, S.K. Udgata and A. Abraham, "Artificial Bee Colony Algorithm for Small Signal Model Parameter Extraction of MESFET," *Eng Appl Artif Intell,* vol. In Press, Corrected Proof, 2010.

[140] A. M. Bernardino, E. M. Bernardino, J. M. Sánchez-Pérez, J. A. Gómez-Pulido and M. A. Vega-Rodríguez, "Efficient Load Balancing for a Resilient Packet Ring using Artificial Bee Colony," *Applications of Evolutionary Computation,* pp. 61-70, 2010.

[141] Y. Delican, R.A. Vural and T. Yildirim, "Artificial Bee Colony Optimization Based CMOS Inverter Design Considering Propagation Delays," *in Symbolic and Numerical Methods, Modeling and Applications to Circuit Design (SM2ACD), 2010 XIth International Workshop on,* pp. 1-5, 2010.

[142] M. Abachizadeh, M.R.H. Yazdi and A. Yousefi-Koma, "Optimal Tuning of PID Controllers using Artificial Bee Colony Algorithm," *in Advanced Intelligent Mechatronics (AIM), 2010 IEEE/ASME International Conference on,* pp. 379-384, 2010.

[143] R.K. Jatoth and A. Rajasekhar, "Speed Control of PMSM by Hybrid Genetic Artificial Bee Colony Algorithm," *in Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference on,* pp. 241-246, 2010.

[144] D. Karaboga, S. Okdem and C. Ozturk, "Cluster Based Wireless Sensor Network Routings using Artificial Bee Colony Algorithm," *in Autonomous and Intelligent Systems (AIS), 2010 International Conference on,* pp. 1-5, 2010.

[145] Hsin-Chih Wang, Yu-Cheng Wang and Men-Shen Tsai, "Performance Comparisons of Genetic Algorithm and Artificial Bee Colony Algorithm Applications for Localization in Wireless Sensor Networks," *in System Science and Engineering (ICSSE), 2010 International Conference on,* pp. 469-474, 2010.

[146] Q. Pan, M. Fatih Tasgetiren, P.N. Suganthan and T.J. Chua, "A Discrete Artificial Bee Colony Algorithm for the Lot-Streaming Flow Shop Scheduling Problem," *Inf. Sci.,* vol. In Press, Corrected Proof, 2010.

[147] M.F. Tasgetiren, Quan-Ke Pan, P.N. Suganthan and A.H.-. Chen, "A Discrete Artificial Bee Colony Algorithm for the Permutation Flow Shop Scheduling Problem with Total Flowtime Criterion," *in Evolutionary Computation (CEC), 2010 IEEE Congress on,* pp. 1-8, 2010.

[148] M.F. Tasgetiren, Q. Pan, P.N. Suganthan and A.H. Chen, "A Discrete Artificial Bee Colony Algorithm for the Total Flowtime Minimization in Permutation Flow Shops," *Inf. Sci.,* vol. In Press, Uncorrected Proof, 2011.

[149] R. Venkata Rao and P.J. Pawar, "Parameter Optimization of a Multi-Pass Milling Process using Non-Traditional Optimization Algorithms," *Applied Soft Computing,* vol. 10, no. 2, pp. 445-456, 2010.

[150] V.K. Vishwa, F.T.S. Chan, N. Mishra and V. Kumar, "Environmental Integrated Closed Loop Logistics Model: An Artificial Bee Colony Approach," *in Supply Chain Management and Information Systems (SCMIS), 2010 8th International Conference on,* pp. 1-7, 2010.

[151] M. Sonmez, "Artificial Bee Colony Algorithm for Optimization of Truss Structures," *Applied Soft Computing,* vol. 11, no. 2, pp. 2406-2418, 2011.

[152] S.N. Omkar, J. Senthilnath, R. Khandelwal, G. Narayana Naik and S. Gopalakrishnan, "Artificial Bee Colony (ABC) for Multi-Objective Design Optimization of Composite Structures," *Applied Soft Computing,* vol. In Press, Corrected Proof, 2009.

[153] R.V. Rao and V.K. Patel, "Optimization of Mechanical Draft Counter Flow Wet-Cooling Tower using Artificial Bee Colony Algorithm," *Energy Conversion and Management,* vol. 52, no. 7, pp. 2611-2622, 2011.

[154] The General Algebraic Modeling System, "GAMS," www.games.com.

[155] W. Yeh and T. Hsieh, "Solving Reliability Redundancy Allocation Problems using an Artificial Bee Colony Algorithm," *Comput. Oper. Res.,* vol. 38, no. 11, pp. 1465-1473, 2011.

[156] P. Jackson, Introduction to Expert Systems, Addison Wesley, 1998.

[157] F. S. Abu-Mouti and M.E. El-Hawary, "Optimal DG Placement for Minimizing Power Loss in Distribution Feeder Systems using Sensory-Deprived Optimization Algorithm," *in Electrical and Computer Engineering (CCECE), 2011 24th Canadian Conference on,* pp. 205-209, 2011.

[158] D.C. Walters and G.B. Sheble, "Genetic Algorithm Solution of Economic Dispatch with Valve Point Loading," *Power Systems, IEEE Transactions on,* vol. 8, no. 3, pp. 1325-1332, 1993.

[159] J.H. Talaq, F. El-Hawary and M.E. El-Hawary, "A Summary of environmental/economic Dispatch Algorithms," *Power Systems, IEEE Transactions on,* vol. 9, no. 3, pp. 1508-1516, 1994.

[160] I.A. Farhat and M.E. El-Hawary, "Dynamic Adaptive Bacterial Foraging Algorithm for Optimum Economic Dispatch with Valve-Point Effects and Wind Power," *Generation, Transmission & Distribution, IET,* vol. 4, no. 9, pp. 989-999, 2010.

[161] M.E. El-Hawary, Electrical Power Systems: Design and Analysis, New York: Wiley, 1995.

[162] J. Grainger and W. Stevenson, Power System Analysis, New York: McGraw-Hill, 1994.

[163] H. Saadat, Power Systems Analysis, Boston, Massachusetts: London, WCB/McGraw-Hill, 1999.

[164] M. Fesanghary and M.M. Ardehali, "A Novel Meta-Heuristic Optimization Methodology for Solving various Types of Economic Dispatch Problem," *Energy,* vol. 34, no. 6, pp. 757-766, 2009.

[165] J.C. Smith, "Wind Power: Present Realities and Future Possibilities," *Proceedings of the IEEE,* vol. 97, no. 2, pp. 195-197, 2009.

[166] Global Wind Energy Council (GWEC), vol. 2011, 10/25, 2011. Available: http://www.gwec.net/index.php?id=24

[167] J.H. Talaq, F. El-Hawary and M.E. El-Hawary, "A Sensitivity Analysis Approach to Minimum Emissions Power Flow," *Power Systems, IEEE Transactions on,* vol. 9, no. 1, pp. 436-442, 1994.

APPENDIX

Table A. 1: Generating units' coefficients and characteristics for the 6-unit system.

Units	**1**	**2**	**3**	**4**	**5**	**6**
P^0 **(MW)**	440.00000	170.00000	200.00000	150.00000	190.00000	110.00000
P_{min} **(MW)**	100.00000	50.00000	80.00000	50.00000	50.00000	50.00000
P_{max} **(MW)**	500.00000	200.00000	300.00000	150.00000	200.00000	120.00000
a **($)**	240.0000	200.0000	220.0000	200.0000	220.000	190.0000
b **($/MW)**	7.00000	10.00000	8.50000	11.00000	10.5000	12.0000
c **($/MW^2)**	0.00700	0.00950	0.00900	0.00900	0.00800	0.0075
UR **(MW/h)**	80.00000	50.00000	65.00000	50.00000	50.00000	50.00000
DR **(MW/h)**	120.00000	90.00000	100.00000	90.00000	90.00000	90.00000
Prohibited operating zones	[210,240] [350,380]	[90,110] [140,160]	[150,170] [221,240]	[80,90] [110,120]	[90,110] [140,150]	[75,85] [100,105]

Table A. 2: *B*-loss coefficients' matrix, element, and constant for the 6-unit system.

B_{ij}						
i \ j	**1**	**2**	**3**	**4**	**5**	**6**
1	0.0017000	0.0012000	0.0007000	-0.0001000	-0.0005000	-0.0002000
2	0.0012000	0.0014000	0.0009000	0.0001000	-0.0006000	-0.0001000
3	0.0007000	0.0009000	0.0031000	0.0000000	-0.0010000	-0.0006000
4	-0.0001000	0.0001000	0.0000000	0.0024000	-0.0006000	-0.0008000
5	-0.0005000	-0.0006000	-0.0010000	-0.0006000	0.0129000	-0.0002000
6	-0.0002000	-0.0001000	-0.0006000	-0.0008000	-0.0002000	0.0150000
B_{0i}	-0.0003908	-0.0001297	0.0007047	0.0000591	0.0002161	-0.0006635
B_{00}	0.0056000	--	--	--	--	--

Table A. 3: Generating units' coefficients and characteristics for the 15-unit system.

Units	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
P^0 **(MW)**	400.0000	300.0000	105.0000	100.0000	90.0000	400.0000	350.0000	95.0000
P_{min} **(MW)**	150.0000	150.0000	20.0000	20.0000	150.0000	135.0000	135.0000	60.0000
P_{max} **(MW)**	455.000	455.0000	130.0000	130.0000	470.0000	460.0000	465.0000	300.0000
a **($)**	671.000	574.000	374.000	374.000	461.000	630.000	548.000	227.000
b **($/MW)**	10.100	10.200	8.800	8.800	10.400	10.100	9.800	11.200
c **($/MW^2)**	0.000299	0.000183	0.001126	0.001126	0.000205	0.000301	0.000364	0.000338
UR **(MW/h)**	80.000	80.0000	130.0000	130.0000	80.0000	80.0000	80.0000	65.0000
DR **(MW/h)**	120.000	120.0000	130.0000	130.0000	120.0000	120.0000	120.0000	100.0000
Prohibited	--	[180,225]	--	--	[180,200]	[230,255]	--	--
operating	--	[305,335]	--	--	[305,335]	[365,395]	--	--
zones	--	[420,450]	--	--	[390,420]	[430,455]	--	--

Units	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**--**
P^0 **(MW)**	105.0000	110.0000	60.0000	40.0000	30.0000	20.0000	20.0000	--
P_{min} **(MW)**	25.0000	25.0000	20.0000	20.0000	25.0000	15.0000	15.0000	--
P_{max} **(MW)**	162.0000	160.0000	80.0000	80.0000	85.0000	55.0000	55.0000	--
a **($)**	173.000	175.000	186.000	230.000	225.000	309.000	323.000	--
b **($/MW)**	11.200	10.700	10.200	9.900	13.100	12.100	12.400	--
c **($/MW^2)**	0.000807	0.001203	0.003586	0.005513	0.000371	0.001929	0.004447	--
UR **(MW/h)**	60.0000	60.0000	80.0000	80.0000	80.0000	55.0000	55.0000	--
DR **(MW/h)**	100.0000	100.0000	80.0000	80.0000	80.0000	55.0000	55.0000	--
Prohibited	--	--	--	[30,40]	--	--	--	--
operating	--	--	--	[55,65]	--	--	--	--
zones	--	--	--	--	--	--	--	--

Table A. 4: *B*-loss coefficients' matrix, element, and constant for the 15-unit system.

B_{ij} (i \ j)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.0014	0.0012	0.0007	-0.0001	-0.0003	-0.0001	-0.0001	-0.0001	-0.0003	-0.0005	-0.0003	-0.0002	0.0004	0.0003	-0.0001
2	0.0012	0.0015	0.0013	0.0000	-0.0005	-0.0002	0.0000	0.0001	-0.0002	-0.0004	-0.0004	0.0000	0.0004	0.0010	-0.0002
3	0.0007	0.0013	0.0076	-0.0001	-0.0013	-0.0009	-0.0001	0.0000	-0.0008	-0.0012	-0.0017	0.0000	-0.0026	0.0111	-0.0028
4	-0.0001	0.0000	-0.0001	0.0034	-0.0007	-0.0004	0.0011	0.0050	0.0029	0.0032	-0.0011	0.0000	0.0001	0.0001	-0.0026
5	-0.0003	-0.0005	-0.0013	-0.0007	0.0090	0.0014	-0.0003	-0.0012	-0.0010	-0.0013	0.0007	-0.0002	-0.0002	-0.0024	-0.0003
6	-0.0001	-0.0002	-0.0009	-0.0004	0.0014	0.0016	0.0000	-0.0006	-0.0005	-0.0008	0.0011	-0.0001	-0.0002	-0.0017	0.0003
7	-0.0001	0.0000	-0.0001	0.0011	-0.0003	0.0000	0.0015	0.0017	0.0015	0.0009	-0.0005	0.0007	0.0000	-0.0002	-0.0008
8	-0.0001	0.0001	0.0000	0.0050	-0.0012	-0.0006	0.0017	0.0168	0.0082	0.0079	-0.0023	-0.0036	0.0001	0.0005	-0.0078
9	-0.0001	-0.0002	-0.0008	0.0029	-0.0010	-0.0005	0.0015	0.0082	0.0129	0.0116	-0.0021	-0.0025	0.0007	-0.0012	-0.0072
10	-0.0005	-0.0004	-0.0012	0.0032	-0.0013	-0.0008	0.0009	0.0079	0.0116	0.0200	-0.0027	-0.0034	0.0009	-0.0011	-0.0088
11	-0.0003	-0.0004	-0.0017	-0.0011	0.0007	0.0011	-0.0005	-0.0023	-0.0021	-0.0027	0.0140	0.0001	0.0004	-0.0038	0.0168
12	-0.0002	0.0000	0.0000	0.0000	-0.0002	-0.0001	0.0007	-0.0036	-0.0025	-0.0034	0.0001	0.0054	-0.0001	-0.0004	0.0028
13	0.0004	0.0004	-0.0026	0.0001	-0.0002	-0.0002	0.0000	0.0001	0.0007	0.0009	0.0004	-0.0001	0.0103	-0.0101	0.0028
14	0.0003	0.0010	0.0111	0.0001	-0.0024	-0.0017	-0.0002	0.0005	-0.0012	-0.0011	-0.0038	-0.0004	-0.0101	0.0578	-0.0094
15	-0.0001	-0.0002	-0.0028	-0.0026	-0.0003	0.0003	-0.0008	-0.0078	-0.0072	-0.0088	0.0168	0.0028	0.0028	-0.0094	0.1283
B_{0i}	-0.0001	-0.0002	0.0028	-0.0001	0.0001	-0.0003	-0.0002	-0.0002	0.0006	0.0039	-0.0017	0.0000	-0.0032	0.0067	-0.0064
B_{00}	0.0055	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Table A. 5: Generating units' coefficients and characteristics for the 5-unit system including the *emission's* coefficients.

Units	**1**	**2**	**3**	**4**	**5**
P^0 **(MW)**	50.71180	40.90040	100.09300	116.89430	161.04310
P_{min} **(MW)**	10.00000	20.00000	30.00000	40.00000	50.00000
P_{max} **(MW)**	75.00000	125.00000	175.00000	250.00000	300.00000
a **($)**	25.0000	60.0000	100.0000	120.0000	40.0000
b **($/MW)**	2.0000	1.8000	2.1000	2.0000	1.8000
c **($/MW^2)**	0.0080	0.0030	0.00120	0.0010	0.00150
d **($)**	100.0000	140.0000	160.0000	180.0000	200.0000
g **(rad)**	0.0420	0.0400	0.0380	0.0370	0.0350
α **(lb/h)**	*80.0000*	*50.0000*	*60.0000*	*45.0000*	*30.0000*
β **(lb/MW)**	*−0.805*	*−0.555*	*−1.355*	*−0.600*	*−0.555*
γ **(lb/MW^2)**	*0.0180*	*0.0150*	*0.01050*	*0.00800*	*0.01200*
η **(lb)**	*0.6550*	*0.57730*	*0.49680*	*0.48600*	*0.50350*
δ **(lb/MW)**	*0.02846*	*0.02446*	*0.02270*	*0.01948*	*0.02075*
UR **(MW/h)**	30.00000	30.00000	40.00000	50.00000	50.00000
DR **(MW/h)**	30.00000	30.00000	40.00000	50.00000	50.00000

Table A. 6: Load's demand per hour for the 5-unit system.

Hour (h)	P_D **(MW)**	**Hour (h)**	P_D **(MW)**
1	410.00	**13**	704.00
2	435.00	**14**	690.00
3	475.00	**15**	654.00
4	530.00	**16**	580.00
5	558.00	**17**	558.00
6	608.00	**18**	608.00
7	626.00	**19**	654.00
8	654.00	**20**	704.00
9	690.00	**21**	680.00
10	704.00	**22**	605.00
11	720.00	**23**	527.00
12	740.00	**24**	463.00

Table A. 7: *B*-loss coefficients' matrix for the 5-unit system.

B_{ij} i \ j	1	2	3	4	5
1	0.000049	0.000014	0.000015	0.000015	0.000020
2	0.000014	0.000045	0.000016	0.000020	0.000018
3	0.000015	0.000016	0.000039	0.000010	0.000012
4	0.000015	0.000020	0.000010	0.000040	0.000014
5	0.000020	0.000018	0.000012	0.000014	0.000035

Table A. 8: Value of 10% renewable source power available for case 2-B.

Hour (h)	RS (MW)	Hour (h)	RS (MW)
1	41.00	**13**	70.40
2	43.50	**14**	69.00
3	47.50	**15**	65.40
4	53.00	**16**	58.00
5	55.80	**17**	55.80
6	60.80	**18**	60.80
7	62.60	**19**	65.40
8	65.40	**20**	70.40
9	69.00	**21**	68.00
10	70.40	**22**	60.50
11	72.00	**23**	52.70
12	74.00	**24**	46.30

Table A. 9: Value of 15% renewable source power available for case 3-B.

Hour (h)	RS (MW)	Hour (h)	RS (MW)
1	61.50	**13**	105.60
2	65.25	**14**	103.50
3	71.25	**15**	98.10
4	79.50	**16**	87.00
5	83.70	**17**	83.70
6	91.20	**18**	91.20
7	93.90	**19**	98.10
8	98.10	**20**	105.60
9	103.50	**21**	102.00
10	105.60	**22**	90.75
11	108.00	**23**	79.05
12	111.00	**24**	69.45

Table A. 10: Value of 20% renewable source power available for case 4-B.

Hour (h)	RS (MW)	Hour (h)	RS (MW)
1	82.00	**13**	140.80
2	87.00	**14**	138.00
3	95.00	**15**	130.80
4	106.00	**16**	116.00
5	111.60	**17**	111.60
6	121.60	**18**	121.60
7	125.20	**19**	130.80
8	130.80	**20**	140.80
9	138.00	**21**	136.00
10	140.80	**22**	121.00
11	144.00	**23**	105.40
12	148.00	**24**	92.60