REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

POWER SYSTEM OPTIMIZATION USING JAYA TECHNIQUE TO SOLUTION OF OPTIMAL POWER FLOW

Habeeb Fadhil Kareem ALSAILAWI

Master's Thesis

Supervisor

Prof. Dr. Kevork MARDIKYAN

Istanbul, 2022

POWER SYSTEM OPTIMIZATION USING JAYA TECHNIQUE TO SOLUTION OF OPTIMAL POWER FLOW

Habeeb Fadhil Kareem ALSAILAWI

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2022

The thesis titled POWER SYSTEM OPTIMIZATION USING JAYA TECHNIQUE TO SOLUTION OF OPTIMAL POWER FLOW prepared by HABEEB FADHIL KAREEM ALSAILAWI and submitted on 27/12/2022 has been **accepted unanimously** for the degree of Master of Science in Electrical and Computer Engineering.

____

Prof. Dr. Kevork MARDIKYAN

the Supervisor

Thesis Defense Committee Members:

Prof. Dr. Kevork MARDIKYAN	Department of Electrical Engineering, Altınbaş University	__________________
Asst. Prof. Mesut ÇEVIK	Department of Computer Engineering, Altınbaş University	__________________
Prof. Dr. Omer USTA	Department of Electrical Engineering, Istanbul Technical University	__________________

I hereby declare that this thesis meets all format and submission requirements of a Master’s thesis.

Submission data of the thesis to Institute of Graduate Studies: ____/____/____

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Habeeb Fadhil Kareem ALSAILAWI

Signature

ABSTRACT

POWER SYSTEM OPTIMIZATION USING JAYA TECHNIQUE TO SOLUTION OF OPTIMAL POWER FLOW

Alsailawi, Habeeb Fadhil Kareem

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Prof. Dr. Kevork MARDIKYAN

Date: 12/2022

Pages: 45

Technology that can handle even the most complicated issues in electrical power systems has been developed thanks to extensive scientific study. Many algorithms and technologies have been developed by scientists to enhance power system variables and thereby decrease fuel costs. In this study, we show how to use the JAYA algorithm, a novel optimization strategy, to solve OPF issues. In the experiment, three objective functions of OPF solutions are considered: minimizing fuel costs, minimizing active power losses, and improving voltage stability. After deciding on a set of parameters for the optimization algorithm, the classic power flow program is executed to determine that function. Since we've reached the maximum number of iterations, the algorithm will begin the iterative process, displaying a comparison between the starting values of the description variables and the values generated by the optimization algorithm. Optimal values for active power losses also emerge over time.

Keywords: Power System, Jaya Algorithm, Optimal Power Flow, Minimize Fuel Cost, And Active Power Flow Minimize.

TABLE OF CONTENTS

Pages

LIST OF TABLES

Pages

LIST OF FIGURES

Pages

ABBREVIATIONS

GA : Genetic Algorithm

PSO : Particle Swarm Optimization

NE : Send Node

NR : Received Node

1. INTRODUCTION

1.1 MOTIVATION AND GENERAL REVIEW

When it comes to running the nation's power grid, Power Flow Optimization (OPF) solutions are indispensable [1,2]. It's a smart power flow that makes use of optimization algorithms to arrange control settings for the power grid in the most efficient way possible, taking into account a wide range of constraints [3,4]. Several traditional optimization techniques have been applied to the OPF issue, including nonlinear programming [5,6], Newton's algorithm [7], quadratic programming [8], and decomposition algorithms [9]. In [10], we find a thorough analysis of the (classic) deterministic optimization algorithms that have come before. While these methods can sometimes lead to a global optimization, they also have some drawbacks, such as a high sensitivity for primary research points, an inability to deal with undifferentiated target functions, and getting stuck in local optima (unsafe convergence properties). As an added downside, these algorithms are not able to ensure a complete answer. Therefore, it is essential to propose alternate approaches to overcoming the aforementioned flaws.

Due to recent advances in computing power, OPF problems are increasingly being tackled with optimization strategies drawn from the natural world. Numerous stochastic optimization methods, such as the genetic algorithm [11–13], particle swarm optimization [2], differential evolution [14,15], symmetric search [15,16], Native bee colony algorithm [4,16], gravity search [17,18], distributed algorithm [19,20], and biogeography-based optimization [21,22], have been proposed and implemented to address OPF issues. In [21], a comprehensive look at random search algorithms for solving OPF variables is provided. Since these algorithms avoid getting tangled up in local optima, they are superior to nonlinear OPF at finding global solutions to a variety of problems. The solution space is rapidly and simultaneously evaluated at high speeds to achieve this property. It is well-suited to large-scale optimization problems due to its comprehensiveness and ease of implementation. Integer and dashed variables are no problem for these procedures. While each of these population-based optimization algorithms has its advantages, improper tuning of the algorithm control parameters can either increase the computational burden (by altering the convergence property) or result in a less than optimal solution. Rao [22, 2016] proposed the JAYA algorithm, one of the recently developed population-based optimization methods, to fix this issue. The optimization process for the JAYA algorithm does not involve modifying any algorithm-specific control parameters in contrast to other population-based methods. We've already established that maintaining command of these variables isn't a walk in the park. Using this function, the JAYA algorithm's optimization process can be sped up and its benefits maximized by doing away with the hassle of regulating these parameters by hand. Also, this method requires little in the way of complicated programming or implementation. The technique takes its cue from the optimization principle, which states that any effort to solve a problem should be directed toward finding the best possible answer and away from any that are less than ideal. Consequently, in contrast to many population-based optimization algorithms, the JAYA algorithm has the benefit of avoiding trapping in local optima. The Gaya method outperforms other approaches to Optimization based on sampling populations thanks to this remarkable property. Based on a comparison of 13 benchmark constrained functions obtained using the JAYA algorithm, it was found in [22] that many other algorithms perform better than JAYA at locating the expected global optimal solutions. Many things prompted us to do this study. First, research into using the JAYA algorithm to find optimal energy flow is lacking. Second, as reported in [4,20,23], many population-based algorithms generate non-feasible solutions to many varieties of OPF problems due to the fact that they go against the constraints of operational variables. Lastly, the above algorithms for solving the OPF problem did not take into account the impact of DG. It is therefore crucial to employ a robust optimization algorithm capable of solving the OPF problem both with and without the presence of DG. This article proposes and presents for the first time how Jaya's algorithm can be used to solve a variety of OPF problems, which will help advance the field of OPF solving. A proposed new procedure based on GIA for handling the OPF solution is the most significant improvement. Furthermore, we present a refined formulation of the OPF problem that accounts for DG's impact. The proposed OPF formulation makes use of a larger number of state variables. These include reactive power injection for shunt capacitors, active power generation outputs, real power generation from DGs, regulation of transformer tuning, voltage tuning at generation nodes, and active power generation outputs. This study examined the trade-offs between optimizing for low generation costs, low real power loss, and high voltage stability. The loss sensitivity and generation cost of active and reactive power injection are two key factors in determining the optimal DG mode in meshed networks, which are discussed in depth in this article. Jaya is tested, validated, and demonstrated on the modified IEEE 30-bus networks, which represent sections of the American electric power system (in the Midwestern US).

1.2 POWER SYSTEM OPTIMIZATION

The power system can be broken down into three major components: the power generators, the transmission lines that carry electricity from power plants to load centres, and the distribution networks that bring electricity to individual homes and businesses. The components of a basic power system are depicted in Figure 1.1.

The Power Plant The primary component of any power grid is the generating units. A variety of energy sources, including batteries, fuel cells, and solar panels, can provide direct current (DC) electricity. In a power plant, a turbo generator device uses a rotor spinning in a magnetic field to generate alternating current (AC) power. Turbine rotors have been turned by everything from superheated steam generated from fossil fuels like coal, gas, and oil to the water itself (hydroelectricity) and even wind energy (wind power). Even nuclear power typically uses a nuclear reaction to boil water into steam. The frequency of alternating current produced by a scalar generator is set by the rate at which the poles are combined. The system's generators all spin at the same rate (synchronously), and the desired frequency will be 50 hertz (Hz) in China and Europe and 60 hertz (Hz) in the United States. Generators need more torque to spin at that speed if the system is overloaded, which means more steam must be supplied to the turbines driving them in a conventional power plant. Therefore, the amount of electrical energy supplied is directly proportional to the amount of steam used and the amount of fuel consumed.

A transformer is used when AC power needs to be transferred from one coil to another. The transformer coils in a static transformer are used to transfer electrical energy from one circuit to another via inductive coupling. The changing current in the first or primary winding creates a changing magnetic field for a variable magnetic flux transformer through the secondary winding. The secondary winding experiences a varying electromagnetic force (EMF) or "voltage" as a result of the varying magnetic field. Mutual induction describes this phenomenon. Transformers efficiently change the voltage and current, which is necessary for a wired transmission system to be practical. The ratio of the number of turns in the secondary winding to the number of turns in the primary winding, $\mathrm{Ns/Np}$, determines the magnitude of the induced voltage in the secondary winding, $\mathrm{Vs/Vp}$, in an ideal transformer.

$$\frac{V_s}{V_p} = \frac{N_s}{N_p} \#(1.1)$$

Overhead power lines, which are electrical transmission lines suspended by towers or utility poles, are one type of transmission line used to move energy from generators to consumers. Since the air acts as a natural insulator, transmitting electricity via transmission lines strung above the ground is typically the most cost-effective option. Ground support towers can be made from a variety of materials, including steel (in the form of latticework or tubular columns), concrete, aluminium, reinforced plastic, and occasionally even freshly cut wood. Copper wire is used for medium voltage distribution and low voltage connections to customer premises, but aluminium wire (plain, steel reinforced, or sometimes composite material) is used for the majority of the bare wire conductors in the transmission line. The resistance of a body with a constant cross-section increases with the square of its length and decreases as a function of the square of its cross-sectional area. Except for superconductors, which have no resistance at all, every material has some. For any given body, resistance is measured as the potential difference between its two ends divided by the current flowing through it.

$$R = \frac{V}{I} \#(1.2)$$

Loads, also known as power consumptions, are the end-use consumers of electrical energy. Typical examples of loads include anything from automobiles to factory equipment. The term "load curve" is used to describe the variation in power consumption or network current as a function of time for any given power system. The rated power of the users is used to calculate the loads, and these loads are random variables that can take on a variety of values. For any given load, set of loads, or system, the true force P is defined to be:

$$P = S \cos \phi \#(1.3)$$

One way to store electrical charge is in a capacitor, which was formerly known as a capacitor. However, all commonly used capacitors have at least two conductors that are separated by a non-conductor. In electrical systems, for instance, capacitors are made of metallic flakes separated by a granular layer. Since the dielectric takes some time to charge up to full voltage, the capacitor's associated current is what ultimately determines the voltage. The voltage is therefore said to be driven by the current in the capacitor. Capacitance is measured in units called farads.

An electrical component's instantaneous power input or output can be written as follows:

$$P = \frac{dE}{dt} = \frac{dE}{dQ} = \frac{dQ}{dt} = VI \#(1.4)$$

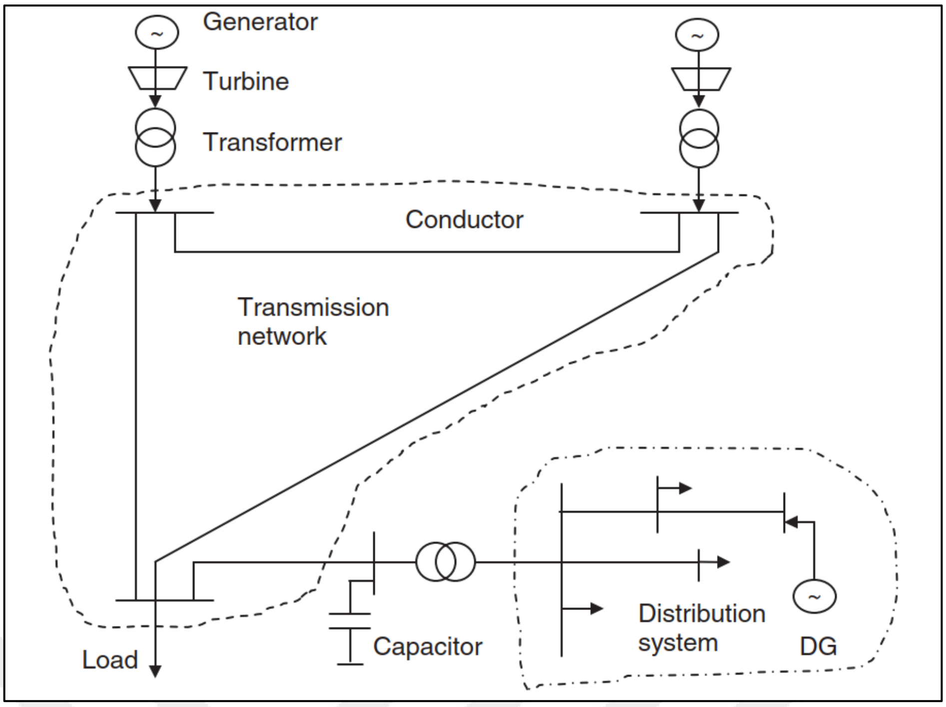

Figure 1.1: Components of power system.

1.3 INTRODUCTION TO OPTIMAL POWER FLOW

Carpentierin 1962 initially presented the OPF. When applied to a grid-based power system, the OPF seeks to identify configurations that simultaneously satisfy constraints on power flow and operating equipment while optimizing the systems objective functions, such as total generation cost, system loss, bus voltage deviation, generation unit emission, number of control procedures, and load separation. In order to achieve the optimal network configuration, it is necessary to adjust a number of control variables, including some of the actual output and voltage of the generators, as well as the pressure change settings of the transformers, phase shifters, transformer capacitors, and reactors.

The OPF problem can be formulated mathematically in various ways, depending on the Objective Functions and constraints used. Generally speaking, they fall into the following categories:

a. A linear problem is one in which the objectives and constraints are presented in linear forms and the control variables are continuous.

b. To clarify, the problem is nonlinear with continuous control variables if either the objective sorcon straints or the two together are nonlinear.

c. Control variables can be either continuous or discrete in mixed-integer linear problems.

An assortment of methods was devised to address the OPF issue. The algorithms can be broken down into three categories: a. traditional a optimization techniques, b. intelligence search techniques, and c. a nonquantitative method of dealing with unknowns in the form of objectives and constraints.

1.4 PROBLEM STATEMENT AND CONTRIBUTION

The mathematical model can be used to formulate the OPF (optimal power flow) problem in either polar or rectangular coordinates. All of the power flows, both real and reactive, as well as the magnitude and phase angle of the voltage at each bus, is represented by polar coordinates. There are no local solutions that are independent of the other parameters when the angle is varied by a multiple of 2. Both the actual and fictitious bus voltages are displayed in a rectangular coordinate system.

Our objectives can therefore be restated as follows:

a. To deal with the OPF issue, an original Jaya-inspired methodology must be developed.
b. Analysing and testing the Jaya algorithm performance on industry-standard power systems: the IEEE 39-bus test grid.
c. The Jaya algorithm addresses all these objective OPF cases in this research: reducing fuel costs, decreasing real power losses and increasing voltage stability.
d. Thus, our choice of JAYA appears to be reasonable in terms to solve the problem.

Finding the best solution has contributed to the lack of studies. There is no objective way to approach optimal energy flow. The application of the Jaya algorithm to various OPF problems has been proposed and presented for the first time in this article as a contribution to the OPF-solving field. Thus, proposing a new Jaya-based procedure for dealing with OPF is the most significant contribution. In addition, a modified version of the (OPF) problem is presented that considers the (DG) effect. For this reason, the OPF formula uses an additional set of state variables. A significant contribution to be made in this article is in the field of optimal placement (DG) in nested networks, which takes into account the loss sensitivity and generation cost of active and reactive power injection. The shunt capacitive power injection is also included in the active powertrain output package. One improvement in this study aims to reduce generation costs, reduce active power loss, and improve voltage stability to take advantage of this advantage.

2. LITERATURE REVIEW

2.1 INTRODUCTION

Currently, optimal power flow (OPF) is a contentious issue. The optimum power flow (OPF) model suggests that the most efficient way to cut costs at electric power plants is to run them at their maximum capacity. Electrical power flows are affected by nonlinear and non-convex functions of the physical properties of the system, making this a challenging problem to solve. However, every five minutes for several independent system operators, the entire distribution network must be resolved in order to meet demand precisely.

2.2 RELATED WORK

The OPF problem is multi-constrained, non-linear, and non-contact, which makes it difficult to solve. Many different methods have been used to solve this problem, including linear programming [24,25], quadratic programming [26,27], simplified gradient method [27,28], Newton's method [29,30], and the inner point method. [31]. The algorithms easily converge to the local optimum when the position of the initial points falls within the convergence range of the optimal local points, which makes them ideal for online computation but not suitable for discrete-variable optimization problems, where the optimization results are closely related to the position of the initial points. The OPF problem can now be tackled with the help of innovative algorithms called experimental algorithms. Common deductive methods consist of things like genetic algorithm (GA) [30,31] and particle swarm optimization (PSO) [32,33]. These algorithms are more efficient at solving discrete-variable problems, have superior global search capabilities, and are not restricted by the location of the starting point. GA has complex encoding and decoding processes to counter this and rapidly converges with local Optima. In addition, the PSO algorithm experiences an early end phenomenon, and the convergence velocity decreases with the lengthening of the evolution time horizon. One recently proposed experimental algorithm is the Artificial Bee Colony (ABC) algorithm, which Karaboga initially proposed in 2005. It approaches how honeybee swarms search for food [34]. It has been successfully used to solve various optimization problems thanks to its effectiveness, scalability, and simplicity. Based on the results of the performance test conducted using some representative numerical functions, it was determined that the ABC algorithm is superior to the GA algorithm, the PSO algorithm, and the DE algorithm when it comes to solving numerical optimization problems [35]. The optimum reactive power flow problem with active power loss minimization where the optimization target was solved using the ABC algorithm in [36], with simulations showing that the ABC algorithm produces less active power loss than the discovery algorithms. The convergence properties of the heuristic or ABC algorithm are superior. The power loss obtained by the ABC algorithm is lower and the running time is shorter than other algorithms, as evidenced by the simulation results in [37], where the ABC algorithm was used to solve the problem of reconfiguring the power system with active power loss as the target function. The ABC algorithm's exploration ability is superior to other algorithms when applied to the economic transfer problem in [38]. While the ABC algorithm has shown promise in preserving and benefiting from population diversity [39, 40], it still has room for development in these areas.

Genetic Algorithm (GA), Ant Colony Optimization Algorithm (ACO), Researcher Optimization Algorithm (ROA), Learning Based Optimization (TLBO), Firefly Algorithm (FA), Crow Search Algorithm (CSA), Differential Evolution (DE), Trace Regressive search optimization (BSO) and particle swarm optimization (PSO) are just a few of the approaches that have been developed in recent years to address complex and nonlinear problems (PSO). As a solution to the optimal reactive power transmission (ORPD) problem, we present a two-point estimation approach. The ORPD issue has been proposed to be addressed by a general-purpose enhancer for the Gray wolf. In order to efficiently solve the ORPD problem, we present a method for minimizing the PL by employing the moth-flame optimization technique, and we optimize the Gravity Search Algorithm (GSA) to deal with constraints. With a Gaussian minimal WCA, you should aim to minimize PL and VD (NGBWCA). ORPD problems with a single or multiple objectives can be solved with the help of the Backward Search Algorithm (BSA). By decreasing (VD) and monitoring the corresponding PL and vice versa, the cuckoo search algorithm is able to solve the multi-objective ORPD problem. Because of this uncertainty, an enhanced version of the firefly algorithm was used to generate a multi-purpose ORPD solution. In this paper, we present the Opposition Krill Herd Algorithm (OKHA) for solving an ORPD problem with a single objective function and MOF in order to minimize both PL and VD. In order to control VAR for a variety of purposes, the DE algorithm is implemented. In order to work in tandem with RDG, the Quest for Harmony had to be designed. The ORPD problem is addressed by employing a genetic algorithm combined with fuzzy target programming. Methods of conditional selection are used to successfully address the limitations of the system. The ORPD has been proposed to be solved using fuzzy linear programming techniques. Strategic placement of shunt capacitors has been proposed for their technical and financial merits.

Reducing pollution in the environment has risen in prominence as a pressing problem around the globe. Therefore, a power system operator may want to rethink making cutting down on fuel costs for generation their top priority. As a result, this paper employs the emission target function to lessen harmful emissions into the atmosphere. Unfortunately, doing so will significantly raise the bar for computational complexity in the OPF problem. The price penalty factor can be used to reduce this problem's dimensionality from multiple objectives to a single one. OPF and OPF with emission problems, with or without renewable sources, are solved using a wide range of optimization methods in the literature. The OPF problem has been approached in the literature using a variety of classical techniques, including the gradient method, quadratic programming, nonlinear programming, and inner point methods. Local minima trapping, the curse of dimensionality, and certain theoretical assumptions do not guarantee an optimal solution when using these methods. Objective functions in the OPF problem are typically inhomogeneous, non-convex, and non-differentiable. To this end, it is crucial to find novel approaches to obtaining the globally optimal solution to the OPF problem, especially when the emission target function is taken into account. Therefore, the OPF problem is addressed by employing a variety of heuristic algorithms. An OPF problem with a single objective function can be solved with particle swarm optimization (PSO). While employing a differential search algorithm. In an effort to fix OPF for multiple uses. Some alternatives have been proposed, such as the Shuffled Frog Leaping Algorithm (SFLA) and the Particle Swarm Optimization (PSO). Authors hope to reduce gas prices and pollution by making these changes. To address the emission and non-sequential cost functions in the OPF, he proposed a regressive search optimization algorithm. This paper presents a new OPF problem model that accounts for wind, solar, and solar combined heat.

2.3 DETERMINISTIC METHOD

The optimal power flow problem can be solved using a variety of deterministic (or classical) approaches. By taking into account objectives, equality, and inequality constraints, the right solution can be found to satisfy the power system's needs. The OPF is widely known thanks to Carpentier (1979) and his heuristic approaches. It uses a mathematical programming technique that works for linear and nonlinear problems as well as those with arbitrary sizing constraints on the underlying power system structure. Linear, quadratic, gradient, Newton, and interior point programming are examples of classical approaches. However, the classical approaches have significant drawbacks, including difficulty in computation and getting stuck at local optima.

2.4 LINEAR PROGRAMMING METHOD

The goals and constraints of the power flow problem are expressed as a linear function in the linear programming approach. If the equation can be simplified, the best solution may be more amenable to negotiation. Since DC power flow has linear constraints, additional linearization is unnecessary. A method known as sequential linear programming, first demonstrated by Griffith & Stewart (1961), is a development of the original linear programming. Successive linear programming is another name for this method. By transforming the nonlinear function into a linear one, the problem was fixed. The method has the drawback of needing a more precise solution in order to find the optimal one for nonlinear programming problems. There are several approaches to linear programming, one of which is the Simplex method. Wells (1968) used the Simplex method to find an optimal power flow under unusual circumstances while taking into account multiple criteria, including cost, reliability, and environmental impact. It is a technique for resolving the economic schedule problem by way of an overloaded set of constraints. Discrete-variable problems can be reliably solved using the mixed-integer linear programming approach, which also helps to avoid nonoptimal solutions. Linear programming's strengths lie in its dependability, ability to converge, and simplicity when dealing with inequalities. As Rau (2003) demonstrated, the linear programming approach is ineffective when the global solution cannot be guaranteed. With the help of the mixed integer linear programming method, Shao Chengcheng et al. (2017) determine the optimal power flow problem in a multicarrier energy system. If you have a problem that requires a lot of calculations, it gave you the best possible result.

2.5 GRADIENT METHOD

The gradient method is simple to implement, trustworthy, and guarantees convergence by separating the variables into dependent and independent categories. Using a verdict approach based on Lagrange-Kuhn-Tucker in the gradient method, Shen and Laughton (1969) were able to solve the economic dispatch problem for the 135 kV British system. This approach required less computational time to solve the problem. Salgado et al. (1990) solved four problems to demonstrate the adaptability of this approach regarding analytical formulation. The primary issue with this strategy is that it yields inaccurate optimization answers to the convex problem. In 2016, Gan Lingwen and Steven H. Low proposed an online algorithm to solve radial networks. The computation time for gradients across bus lines is shortened.

2.6 EVOLUTIONARY ALGORITHM

The evolutionary approach takes its cues from the process of natural selection. It has provided the optimal answer to issues that can be tackled without resorting to assumptions about things like repetition, convexity, smoothness, etc. Yuryevich Jason and Kit Po Wong (1999) proposed using gradient information in an evolutionary algorithm to speed up convergence and manage more complex systems. The optimal solution can be reached with this algorithm without specifying a precise gradient step size. The developed algorithm can find the global optimum solution to the OPF problem when there are multiple local optima. This is because of the population of candidate solutions. There is no assurance of finding the optimal solution in a finite amount of time, but Abido (2003) used an evolutionary algorithm to solve the OPF problem and talk about its benefits. An improvement in performance and elimination of the method's drawbacks are possible through tweaking the original design. To solve the environmental and economic challenges of electric power dispatch, Abido (2004) industrialized the strength Pareto evolutionary algorithm. The novel approach was replicated for excellence measurement, and a new procedure for investigating the boundaries of a bounded area was introduced. Artificial intelligence was first developed using evolutionary programming, which Fogel (2006) introduced by simulating evolution as a learning process. The nonlinear constrained multi-objective problem of optimizing the generation of real power while maintaining the voltage stability of the system was proposed and solved by Al-Hajri and Abido (2011) using an enhanced strength Pareto evolutionary algorithm. Only the most important goal was optimized, and the others were treated as constraints with limits.

2.7 MULTI-OBJECTIVE OPTIMIZATION

Many different goals and approaches to finding optimal solutions exist for power system optimization issues. A purely one-sided optimization approach is taken when solving the optimal power flow problem. However, in practical power system issues, optimizing for multiple goals at once is necessary. The multi-objective problem is solved using a variety of algorithms. The voltage security costs could be resolved using the penalty method proposed by Fonseca & Fleming (1995), but this approach presents the challenge of determining an appropriate penalty. When multiple objectives are combined into one, it is unnecessary to maintain the importance of any original objectives. The constraint method was developed to address the shortcoming of the weighted sum approach. Mixed-integer nonlinear programming with a -constrained optimization approach was used to find an optimal solution for multiple objectives related to power consumption. Multi-objective Optimization yields better solutions than single-objective Optimization (Ara et al., 2012), though the exact nature of the improvement depends on the -constraint. The limitations of the penalty method, the weighted sum, and the -constraint motivate the use of the Pareto optimal. The Pareto optimal method is used so that the importance of all goals can be preserved. Using this strategy, we were able to achieve a single outcome by focusing our efforts on the non-dominated outcomes (the Pareto optimal solution). The Pareto optimal solution is computationally efficient, simple to implement, and widely used. The programmer can choose the best option among several viable options, and the decision maker can choose the best option. The Pareto optimal solution is used to solve multi-objective problems, and several optimization techniques supports it. According to Venkatesh et al. (2000), a multi-objective fuzzy LP algorithm is used to maximize the voltage stability margin indicator. The evolutionary algorithm Cai and Wang (2006) used to transform a dominant individual into a non-dominant one solved a constrained multi-objective optimization problem. The capabilities for manipulating the recombination operator are developed using the simplex boundary. The non-dominated sorting genetic algorithm was used by Varadarajan and Swarup (2008) to solve the multi-objective problem (NSGA-II). By controlling the inequality constraints, it also sidesteps the penalty coefficients. Abido and Al-Ali (2009) used differential evolution to achieve multiple competing goals. Clustering algorithm is used to manage the massive number of possible answers to these problems, and fuzzy set theory is used to determine the best answers. This approach has a number of drawbacks, the most significant being its unstable convergence and propensity to quickly settle on a local optimum. By employing the multi-objective harmony search algorithm simultaneously, Sivasubramani & Swarup (2011) were able to optimize fuel cost and emissions simultaneously. This strategy is employed to enhance the Pareto-optimal set with fine-grained distribution. Finding the optimal middle ground required using the fuzzy membership approach. With lively gathering detachment, a non-dominated solution has been organized and placed. Duman Serhat (2012) used the gravitational search algorithm to calculate alternative objective functions under both typical and extreme circumstances. Inadequate initial population generation, a lengthy computation time, and a convergence problem are all drawbacks of this approach. To solve the IEEE 30-bus and 118-bus test systems' NP-hard problems, Khorsandi et al. (2013) used a modified artificial bee colony (MABC) algorithm with a more efficient global search. Getting the right result will take more iterations. The test result demonstrates that the MABC algorithm maintains its performance even as the problem dimension increases. Yuan Xiaohui (2017) used an enhanced strength Pareto evolutionary algorithm to achieve their fuel cost and emission goals. As a result of these enhancements, the standard strength Pareto evolutionary algorithm achieved more evenly distributed Pareto optimal solutions.

2.8 JAYA ALGORITHM

Machine learning, deep learning, knowledge representation, inference, and perception are just a few of the many branches of AI, and one of the most common problems they face is finding the best possible solution in the search space. Empirical, problem-based concepts are used in conventional research methods to find approximate solutions, with less emphasis on quality. Artificial intelligence (AI) researchers have recently become interested in algorithms based on metaheuristics because of their impressive search space addressing capabilities. A flexible optimization framework, the Metaheuristic-based algorithm can be applied to a wide variety of problems. Learning mechanisms that take advantage of the existing body of knowledge and search space exploration are at the heart of its iterative development approach. Parameters are adjusted to find an approximation that works well enough for the optimization problem at hand.

The JAYA algorithm is a cutting-edge metaheuristic-based algorithm that takes the best features of EA (resilience according to the fattest principle) and SI (following the leader to find the best solution) and combines them. Rao proposed the JAYA algorithm in 2016, and ever since then, it has garnered significant interest from a wide range of research communities thanks to its remarkable properties. The data gathered from the first search was absent. This is a parameter less algorithm. It can be moulded in various ways, is reliable, and has all the necessary parts. Therefore, many optimization problems have been solved using the JAYA algorithm, including power flow, parameter extraction for solar cells, bag problems, virtual machine mode, workshop scheduling, permutation shop scheduling problem, reliability, and redundancy. Issues with customization, panel formation, truss design, emotion recognition, feature selection, platen heat exchanger, estimating parameters for a model of a Li-ion battery, etc. Many researchers have tweaked the JAYA algorithm to make it more effective at solving optimization problems with complex constraints and rough properties in their search spaces. Others are enhancing the JAYA algorithm by incorporating features from other powerful optimization algorithms. This is why many different proposed JAYA binary algorithms exist. Hybridization with evolutionary algorithms, swarm intelligence algorithms, physical algorithms, and other components; JAYA self-adaptive algorithm; JAYA-based elitism; JAYA multi-population self-adaptive algorithm; elitism-based; Chaotic JAYA algorithm; JAYA neural network algorithm. This paper provides an in-depth, though not exhaustive, overview of JAYA algorithms from a number of perspectives: Theoretical premises are first discussed and analysed. Next, we take a look at the JAYA algorithm in its various iterations: pure, hybrid, and adapted. JAYA algorithms' coverage of various use cases is briefly described under several headings. With this summary in hand, the reader will be able to pinpoint the domains and uses that have made use of the JAYA algorithm and provide justification for their own contributions to the framework. Moreover, the JAYA algorithm's open-source code has been defined to supply enrichment resources for the JAYA research communities. As a result, this paper will evaluate the JAYA algorithm and highlight its strengths and weaknesses. The current contributions of the JAYA algorithm, possible future directions for performance improvement, and the usefulness of the JAYA algorithm are summarized in the final section.

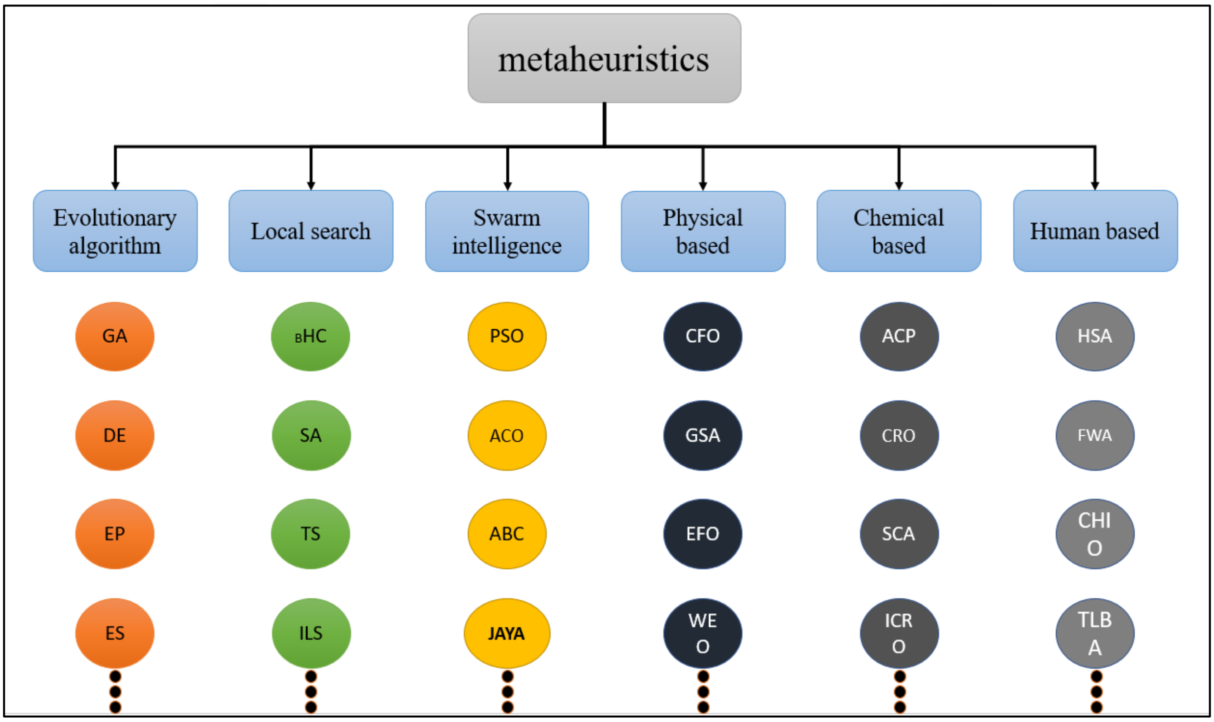

Figure 2.1: Classes of algorithms based on metaheuristics.

2.8.1 Algorithms Based on the JAYA Framework That Have Been Modified

Other experiments have been conducted to enhance the JAYA algorithm's performance to better fit the optimisation problem's search space. In order to enhance wind farm planning, Rao et al. [41] presented a new variant of the JAYA algorithm. The notation for this procedure is the MTPG-Jaya oriented multiband perturbation algorithm. The MTPG-Jaya algorithm uses a swarm of agents guided by a variety of perturbation equations acting on a single population to probe the search space for a solution. Each team promotes the same population. Using what we now know about the most successful groups, we were able to improve upon the disruption equation for the worst performing groups. The solution's value and any violations of its boundaries are now in your possession. Three optimization cases for planning a wind farm are used to assess the efficacy of the MTPGJaya algorithm. The simulation results demonstrated that the MTPG-Jaya algorithm outperformed the baseline JAYA algorithm and its competitors. The MTPG-Jaya adaptive algorithm was introduced in another paper ([42, 43]). By dynamically adjusting the number of teams in response to the algorithm's convergence performance, it is able to intelligently probe more areas of the search space. In order to estimate photovoltaic parameters and module models, Yu et al. [44] proposed a performance-oriented JAYA (PGJAYA) algorithm. Probability is used in PGJAYA to assess how well the community has received a solution. Then, during the investigation phase, each option adaptively chooses among a variety of evaluative mechanisms that aim to strike a balance between exploration and exploitation. Qualifying performance is used to select models that will lead to fruitful research. Last but not least, the best global solutions are swapped out for the worst possible solutions by employing the self-adaptive chaotic perturbation mechanism. This purpose is to raise the standard of living for all population. As can be seen from the simulation results, the PGJAYA algorithm is a reliable means of determining the PV module and cell model parameters. As in [45], a variant of the JAYA algorithm is optimized to maximize the use of renewable energy sources. This method is referred to as MJAYA. To optimize for the community as a whole, this algorithm takes into account the qualities of the best members while ignoring those of the worst. Incorporating this change moves everyone closer to the best person in society. Five case studies of the IEEE 30-bus system and the IEEE 118-bus system are used to assess the effectiveness of the MJAYA algorithm. The MJAYA algorithm outperformed its rivals in terms of solution quality and number of evaluations in both theoretical and practical simulations, including the moth swarm algorithm, the artificial bee colony algorithm, the cuckoo search algorithm, the gray wolf optimizer, and the regressive search optimization algorithm. Many population, including Mani. [46] He also presented a new and improved variant of the JAYA algorithm, which he dubbed "quasi-stable JAYA" (SJaya). The authors of the SJaya algorithm present a novel approach to updating society's best and worst members. This is accomplished by adopting a fresh set of standards by which to evaluate potential new members and weed out those who don't meet the requirements. SJaya is optimized for the fuel cell stack and its efficacy is measured against twelve canonical standard functions. In terms of solution quality and speed at which a near-perfect solution can be found, SJaya was found to perform better than the original version of the JAYA algorithm in the simulation results.

The authors of [47] presented an updated version of the JAYA algorithm for solving multi-objective optimization problems (MaOPs). A many-objective JAYA (MaOJaya) algorithm describes their method of operation. In MaOJaya, both the original and altered populations are counted as part of a single total. Individuals for the subsequent cycle are chosen using the tournament selection scheme. This adjustment aims to speed up the algorithm's search for the ideal candidate. The Tchebychef decomposition method is then applied to the MaOPs for final solution. We use DTLZ benchmark functions with 3, 5, 8, and 10 goals to test the MaOJaya algorithm. Experiments demonstrated that the MaOJaya algorithm outperformed or was competitive with other algorithms in solving the MaOPs. Yu et al. [48] create a variant of the JAYA algorithm they call the oppositional Jaya (OJaya) algorithm. The OJaya algorithm incorporates oppositional learning into the JAYA algorithm to increase population diversity and search space size for a given problem. Individuals in populations are directed toward the optimal solution and away from the worst one using the distance-adaptive coefficient, which is used to identify the best and worst members of the population. On a 3-bus, 8-bus, 9-bus, and 15-bus testing system, we apply the OJaya algorithm to the directions over current relays coordination problem to gauge its efficacy. When compared to the original JAYA algorithm and other similar algorithms, the OJaya algorithm performed better in simulations for all measures of solution quality, convergence speed, robustness, and computational efficiency.

2.8.2 The JAYA Algorithm Theory: A Critical Analysis

JAYA has attracted a lot of attention because it can be applied to many different kinds of optimization problems, as shown by the list of JAYA's variants and applications. This is because of its many desirable qualities: Conceptually straightforward, user-friendly, parameter-free, derivative-free, and mathematically sound and complete. This is the primary cause of the algorithm's success and the reason why academic communities have widely adopted it. However, the JAYA algorithm has the same inherent limitations and drawbacks as other metaheuristic algorithms. The No Free Lunch (NFL) Theorem of Optimization is the main problem with this algorithm. No one best optimization algorithm in the NFL will always come out on top, no matter how many times the same optimization problem is run. Therefore, the characteristics of the problem search space at hand have a direct bearing on the JAYA algorithm's convergence behaviour. As a result, the JAYA algorithm requires some sort of adaptation or hybridization to deal with the problem's search space. The second limitation is related to the nature of the problem area. JAYA's initial proposal for optimization problems was formulated with a single objective, an unrestricted objective function, and a continuous field. However, JAYA needs to broaden its scope of application in order to be effective in other problem areas and types, including discrete, dual, integrative, dynamic, and many, multiple goals and objectives.

Jaya algorithm has a third flaw that has to do with the way it interacts with groups of populations. During the search process, the Jaya algorithm is able to explore a large search space thanks to its one-of-a-kind operator, which allows for the development of research based on inherited values attracted to the best global solutions and the progression to better search space regions via distance expansion with worse solutions. There is a deceptive quality to the study in this way. However, its operator does not pay attention to all parts of a converging search space, leading to a decline in the exploitation process that is especially problematic for issues in the multimedia domain. Thus, to further expand its use, the Jaya algorithm should be hybridized with another algorithm from local research. By destroying so much of Jaya's bridge at once, local search methods can zero in on the most promising sections of the search landscape. To the same extent as other methods of enhancement, the Jaya algorithm's performance is linked to the issue of dimensions. The efficiency of the algorithm breaks down as the number of input variables to the problem increases. As a result, decreasing the number of dimensions in machine learning is an acceptable solution to the issue of some return on more EFCient results. It's important to think about alternative mechanisms for scaling up and down. Therefore, as discussed above, the GWO algorithm outperformed Jaya in some use cases. GWO may even outperform Jaya in high-dimensional problems because of its lower susceptibility to dimensionality issues. Jaya's reliance on truly random randomness in its operator pool is a further widespread issue. Since the artistic periodic nature of false randomness means that some patterns will repeat themselves over time, the research pattern will also adopt a periodic behaviour.

At last, the JAYA algorithm faces a diversity control concordance dilemma. When the initial run's diversity is lost, the JAYA algorithm's convergence behaviour tends to stall at the local minimum. This occurs because the JAYA algorithm's unique operator only takes into account the best and worst solutions in each generation and ignores all other features of the community's solutions. In doing so, he will be drawn to the most advantageous options and repelled by the characteristics of the least desirable ones. It is well-established in the field of Optimization that the path taken in search of the global Optima need not necessarily lead to the optimal answer. Consequently, there is a lack of diversity because the vast majority of the population's solutions are being ignored. Some research suggests that a more effective method of diversity control would be to divide the population into subpopulations and give each subpopulation its own isolated JAYA. In addition, I am relieved to know that using parallel JAYA can drastically accelerate their convergence without compromising solution quality.

3. METHODOLOGY

3.1 INTRODUCTION

In the science and technology area, optimization plays a prominent role. It refers to locating the most satisfactory result of an objective which is used to maximize or minimize while simultaneously satisfying several constraints. The optimization model makes an objective function that may be linear or nonlinear. The constraint corresponds to equality or inequality, which confines decision variables.

3.2 PROPOSED ALGORITHM (JAYA)

With the introduction of the Jaya model, Rao has provided a new population-based optimization that can efficiently solve optimization issues without constraints. Gaia differs from other population-based exploratory algorithms in that it does not include an algorithm control parameter, instead providing only the two standard parameters for adjusting population size (m) (i.e., the candidate solutions number) and the generations number (Gn) (i.e. total iterations). The principle inspires the optimization process for this method that a problem's "best" solution should progress away from "best" minima. In light of the preceding, the fundamental Jaya algorithm is a straightforward optimization method consisting of a single stage. Detailed steps of the Gaia algorithm are depicted in Figure 3.1.

The proposed method for using the Jaya algorithm to address the OPF issue is laid out in the following.

The network details about the branches active power and generating units must be included. Active power losses for buses, reactive power for shunt compensators, voltage magnitudes for generating buses, and tap settings for regulating transformers must all be provided as beginning conditions. Start up the load status dispatching system. This study aims to achieve two main objectives: optimal fuel cost and minimizing active power losses. To begin, will need to determine the target functions values, which include the fuel cost, minimizing active power loss, and boosting voltage stability. Distributed generation (DG) units should be placed at appropriate locations considering the active power loss sensitivity and fuel cost, as described in the following formulations.

$$\begin{bmatrix}\frac{\partial P_{\text{loss}}}{\partial P_i}\\ \frac{\partial P_{\text{loss}}}{\partial Q_K}\end{bmatrix} = [Jac]^{T^{-1}}\begin{bmatrix}\frac{\partial P_{\text{loss}}}{\partial \delta_i}\\ \frac{\partial P_{\text{loss}}}{\partial V_K}\end{bmatrix}\#(3.1)$$

The system parameters are represented as mathematical variables. The active power losses sensitivity divided by active power injected at the bus and reactive power injected at the load bus are denoted by ∂Ploss/∂Pi and ∂Ploss/∂QK, respectively. The active power losses can affect the distribution of power flows along transmission lines and the energy mix throughout power systems. Regarding to active power losses, ∂Ploss/∂δi is the sensitivity to voltage angle of the bus, and ∂Ploss/∂VK is the sensitivity to voltage magnitude of the load bus. In this context, [Jac] is an abbreviation for Jacobian matrix.

$$\begin{bmatrix}\frac{\partial \operatorname{Cos} t}{\partial P_i}\\ \frac{\partial \operatorname{Cos} t}{\partial Q_K}\end{bmatrix} = [Jac]^{T^{-1}}\begin{bmatrix}\frac{\partial \operatorname{Cos} t}{\partial \delta_i}\\ \frac{\partial \operatorname{Cos} t}{\partial V_K}\end{bmatrix}\#(3.2)$$

∂Cost/∂QK represents the generation cost sensitivity to reactive power injection at the k bus, and ∂Cost/∂Pi represents the generation cost to active power injection at the I bus. The term ∂Cost/∂δi denotes the cost of generation sensitivity to the angle of voltage of the I bus. In contrast, the term ∂Cost/∂VK indicates the generation cost sensitivity to the voltage magnitude of the k load bus. This third step is only implemented when the DG effect is considered.

Create an optimization objective function by noting it as Fi(x,u) (one of the objective functions). The max and min values have been Sets for design variables, the iterations number (Gn), and the system variables number (n). Generate an initial sample population using a random generator and the defined control parameters within the bounds of the design variables. The breakdown of this group of population looks like this:

$$\text{population} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n}\\ x_{21} & x_{22} & \cdots & x_{2n}\\ x_{31} & x_{32} & \cdots & x_{3n}\\ \vdots & \vdots & \vdots & \vdots\\ x_{m1} & x_{m2} & \cdots & x_{mn}\end{bmatrix}\#(3.3)$$

$$X_{k,j} = X_j^{\min} + \text{rand}(\quad).\left[X_j^{\max} - X_j^{\min}\right]\#(3.4)$$

$$= \begin{bmatrix} P_{G1,2} \cdots, P_{G1,N_G}, T_{1,1}, \cdots, T_{1,N_T}, V_{G1,1}, \cdots, V_{G1,N_G}, Q_{C1,1,1}, \cdots, Q_{C1,N_C}, P_{DG1,1}, \cdots, P_{DG1,N_{DG}} \\ P_{G2,2}, \cdots, P_{G2,N_G}, T_{2,1}, \cdots, T_{2,N_T}, V_{G2,1}, \cdots, V_{G2,N_G}, Q_{C2,1}, \cdots, Q_{C2,N_C}, P_{DG2,1}, \cdots, P_{DG2,N_{DG}} \\ \vdots \\ P_{Gm,2}, \cdots, P_{Gm,N_G}, T_{m,1}, \cdots, T_{m,N_T}, V_{Gm,1}, \cdots, V_{Gm,N_G}, Q_{Cm,1}, \cdots, Q_{Cm,N_C}, P_{DGm,1} \cdots, P_{DGm,N_{DG}} \end{bmatrix} \#(3.5)$$

The objective function value associated with each solution candidate can be determined by running the power flow program for each solution. Determine which of the possible answers is the best and which is the worst. Adjusting all potential answers in light of the best and worst answers is recommended. Following is an expression of the proposed change.

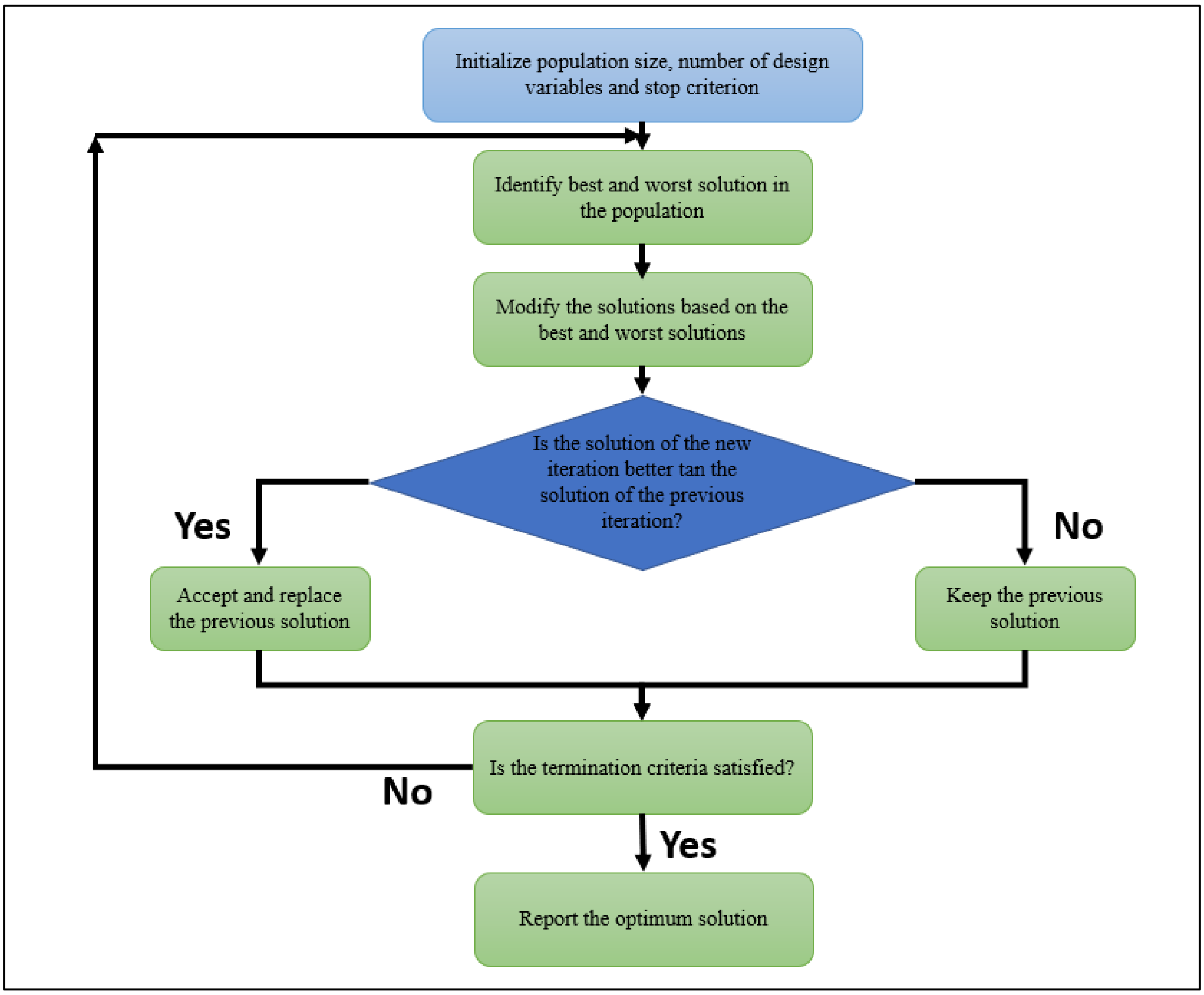

Figure 3.1: Jaya algorithm steps.

$$r_{1,j,i} = R_1, r_{2,j,i} = R_2 \tag{3.6}$$

$$\mathcal{X}'_{j,k.i} = \mathcal{X}_{j,k,i} + R_1[(\mathcal{X}_{j,\text{ best },i}) - (|\mathcal{X}_{j,k,i}|)] - R_2[(\mathcal{X}_{j,worst,i}) - (|\mathcal{X}_{j,k,i}|)] \#(3.7)$$

The second term in the aforementioned equation represents how likely it is that the improved solution will eventually converge on the optimal solution. As implied by its third expression, the solution tends to avoid the worst possible outcome.

In this thesis, we choose to address the OPF problem in two ways: minimizing the total power generation fuel cost and minimizing the active power loss. The following function represents the total fuel cost of a given power network:

$$F_1(x,u) = \mathrm{Cos}\,t = \sum_{i=1}^{N_G} f_i \#(3.8)$$

where NG is the generation units number.

$$f_i = a_i + b_i(P_{Gi}) + c_i(P_{Gi})^2 \left(\frac{\$}{h}\right) \#(3.9)$$

The fuel cost coefficients represent as ai, bi, and ci

Ploss, which is the active power losses, is defined as:

$$F_2(x,u) = P_{\mathrm{loss}} = \sum_{i=1}^{N} \sum_{\substack{i=1 \\ j\neq i}}^{N} \frac{G_{ij}}{2}\left[|V_i|^2 + |V_j|^2 - 2|V_i||V_j|\cos(\delta_i - \delta_j)\right] \#\#\#\#\# \#(3.10)$$

$$j \neq i$$

where N is the network node number, Vi and Vj are the voltage magnitudes.

The following is a rundown of the primary stages involved in the development of the framework:

a. First, will need to adjust some Jaya settings.

b. Secondly, data must be loaded into the system (line and bus data).

c. Produce the initial population within the range of allowable values for the regulating variables.

d. Power flow should be run, and the fitness function for each population solution should be determined.

e. Fifthly, determine which options in the population are the best and worst.

f. Change the population to reflect the best and worst answers.

g. Evaluate the original and new population solutions' fitness function values for each population. Replace the old method with the new one if the improvements warrant it. If this problem cannot be solved, the current solution should be maintained.

h. Stop and report the best possible solution if the stopping criterion is met. But if not, go back to the 5th step.

Tables 3.1, 3.2, and figure 3.2 illustrate the IEEE 39 bus network diagram and the bus and line values.

Table 3.1: IEEE 39 bus data.

No	Voltage	Angle	Pgen	Qgen	Pload	Qload	Gshunt	Bshunt	Type	0,1
1	0.982	0	0	0	0.092	0.046	0	0	1	0
2	1.03	0	10	0	11.04	2.5	0	0	2	0
3	0.9831	0	6.5	0	0	0	0	0	2	0
4	1.0123	0	5.08	0	0	0	0	0	2	0
5	0.9972	0	6.32	0	0	0	0	0	2	0
6	1.0493	0	6.5	0	0	0	0	0	2	0
7	1.0635	0	5.6	0	0	0	0	0	2	0
8	1.0278	0	5.4	0	0	0	0	0	2	0
9	1.0265	0	8.3	0	0	0	0	0	2	0
10	1.0475	0	2.5	0	0	0	0	0	2	0
11	1	0	0	0	0	0	0	0	3	0
12	1	0	0	0	0	0	0	0	3	0
13	1	0	0	0	3.22	0.024	0	0	3	0
14	1	0	0	0	5	1.84	0	0	3	0
15	1	0	0	0	0	0	0	0	3	0
16	1	0	0	0	0	0	0	0	3	0
17	1	0	0	0	2.338	0.84	0	0	3	0
18	1	0	0	0	5.22	1.76	0	0	3	0
19	1	0	0	0	0	0	0	0	3	0
20	1	0	0	0	0	0	0	0	3	0
21	1	0	0	0	2.74	1.15	0	0	3	0
22	1	0	0	0	0	0	0	0	3	0
23	1	0	0	0	2.745	0.8466	0	0	3	0
24	1	0	0	0	3.086	0.922	0	0	3	0
25	1	0	0	0	2.24	0.472	0	0	3	0
26	1	0	0	0	1.39	0.17	0	0	3	0
27	1	0	0	0	2.81	0.755	0	0	3	0
28	1	0	0	0	2.06	0.276	0	0	3	0
29	1	0	0	0	2.835	0.269	0	0	3	0
30	1	0	0	0	6.28	1.03	0	0	3	0
31	1	0	0	0	0	0	0	0	3	0
32	1	0	0	0	0.075	0.88	0	0	3	0
33	1	0	0	0	0	0	0	0	3	0
34	1	0	0	0	0	0	0	0	3	0
35	1	0	0	0	3.2	1.53	0	0	3	0
36	1	0	0	0	3.294	0.323	0	0	3	0
37	1	0	0	0	0	0	0	0	3	1
38	1	0	0	0	1.58	0.3	0	0	3	0
39	1	0	0	0	0	0	0	0	3	1

Table 3.2: IEEE 39 line data.

Line S	Line R	R	X	B	T	0,1
27	37	0.0013	0.0173	0.3216	0	1
37	38	0.0007	0.0082	0.1319	0	1
24	36	0	0.0059	0	1	1
21	36	0.0008	0.0135	0.2548	0	1
36	39	0.0016	0.0195	0.304	0	1
36	37	0.0007	0.0089	0.1342	0	1
35	36	0.0009	0.0094	0.171	0	1
34	35	0.0018	0.0217	0.366	0	1
33	34	0.0009	0.0101	0.1723	0	1
28	29	0.0014	0.0151	0.249	0	1
26	29	0.0057	0.0625	1.029	0	1
26	28	0.0043	0.0474	0.7802	0	1
26	27	0.0014	0.0147	0.2396	0	1
25	26	0.0032	0.0323	0.513	0	1
23	24	0.0022	0.035	0.361	0	1
22	23	0.0006	0.0096	0.1846	0	1
21	22	0.0008	0.0135	0.2548	0	1
20	33	0.0004	0.0043	0.0729	0	1
20	31	0.0004	0.0043	0.0729	0	1
2	19	0.001	0.025	1.2	0	1
18	19	0.0023	0.0363	0.3804	0	1
17	18	0	0.0046	0	1	1
16	31	0.0007	0.0082	0.1389	0	1
16	17	0.0006	0.0092	0.113	0	1
15	18	0.0008	0.0112	0.1476	0	1
15	16	0.0002	0.0026	0.0434	0	1
14	34	0.0008	0.0129	0.1382	0	1
14	15	0.0008	0.0128	0.1342	0	1
13	38	0.0011	0.0133	0.2138	0	1
13	14	0	0.0213	0	1	1
12	25	0.007	0.0086	0.146	0	1
12	13	0.0013	0.0151	0.2572	0	1
11	12	0.0035	0.0411	0.6987	0	1
2	11	0.001	0.025	0.75	0	1
30	39	0.0007	0.0138	0	0	1
5	39	0.0007	0.0142	0	0	1
32	33	0.0016	0.0435	0	0	1
31	32	0.0016	0.0435	0	0	1
4	30	0	0.018	0	1	1
9	29	0.0008	0.0156	0	0	1
8	25	0	0.0232	0	1	1
7	23	0.0005	0.0272	0	0	1
6	22	0	0.0143	0	1	1
3	20	0	0.02	0	1	1
1	16	0	0.025	0	1	1
10	12	0	0.0181	0	1	1

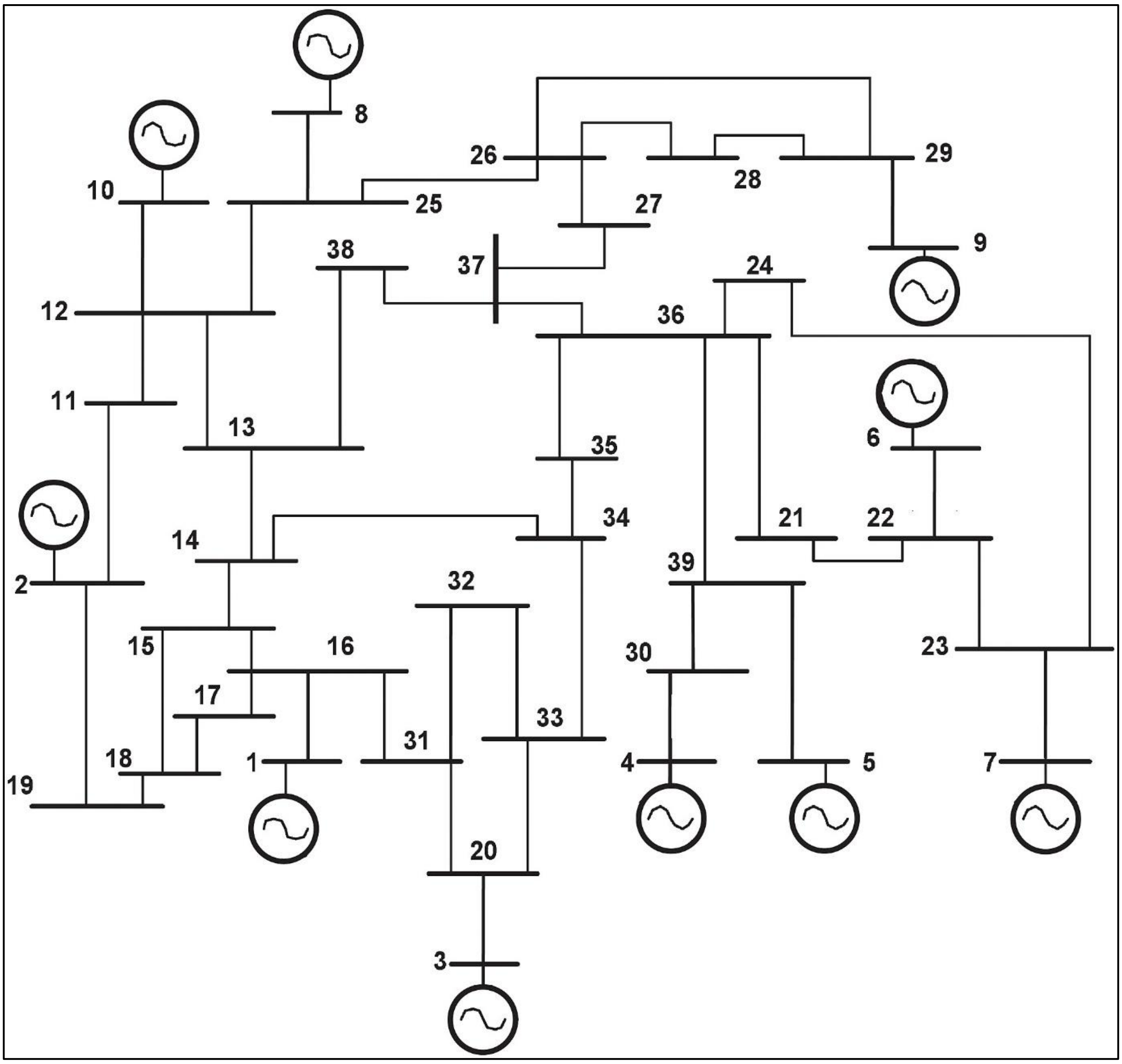

Figure 3.2: IEEE 39-Bus system.

4. RESULTS AND DISCUSSION

4.1 THE EXPERIMENTS

regarding transmission and distribution systems, distribution systems are typically comprised of much more complicated basic ingredients including devices, To obtain precise and trustworthy results from the analysis, those components need to be modelled in as much detail as possible. This section provides an overview of the comprehensive models of distribution system components and devices typically used in these kinds of systems. These models are separated into two distinct categories: branch elements and nodal injection elements.

In this experiment, the proposed model (JAYA) is put through its steps on the IEEE bus standard testing system to demonstrate its effectiveness and accuracy. Technically, researchers use IEEE bus systems to put their theories and concepts into practice and to test their proposed models. The IEEE 39-bus system is an IEEE standard bus system with fixed specifications. Information and details about lines and buses for the IEEE 39-bus system can be found in Chapter 3. The JAYA model is used to solve the problems of optimal power flow to obtain an optimal solution to the fuel cost and to minimize active power loss while optimizing the amount of voltage for an IEEE 39-bus system by optimizing system parameters. This system (IEEE 39-bus) is used in this thesis to put the proposed JAYA through scalability testing and demonstrate the proposed system's ability to deal with large-scale problems. There are ten different generators in the IEEE 39-bus system. Buses connected to generator units are allowed with a voltage between 0.95 and 1.1 p.u.

In contrast, their bounds are between 0.93 and 1.06 p.u. for the buses not connected with the generating units. It is necessary to determine the values of many parameters to obtain the best possible result when implementing the proposed model (JAYA) algorithm. In this thesis, It is via empirical testing, using a variety of parameter settings for the proposed JAYA algorithm, that the optimal values for its parameters are found.

With the development of practical research in improvement algorithms, scientists and researchers have developed many improvement algorithms that have not yet been applied to many problems related to energy systems from them (OPF). The OPF problem was solved by using several methods and improvements to improvement in the literature that we mentioned. The results achieved satisfaction at the scientific level due to the available classic improvement algorithms. Our proposed model is called the JAYA algorithm. The IEEE 39-

bus system is the system will use in our experiments for the test cases. The test cases use data from the standard for the bus and line data. The program that was utilized is called MATLAB, and the number of iterations that were performed for each of the cases ranges between 35 and 100. The figures and tables present the results of the analysis performed on all of the issues.

Table 4.1: Original and JAYA results with 35 iterations.

	VOLTAGE		ANGLE	
Bus	Orig	JAYA	Orig	JAYA
1	0.982	1.048926	0	0
2	1.03	1.034975	-10.8515	-8.68199
3	0.9831	1.05	2.426817	2.268704
4	1.0123	1.00463	3.43347	4.101846
5	0.9972	1.05	4.369007	4.885974
6	1.0493	1.05	5.364398	6.250819
7	1.0635	1.05	8.158688	9.357989
8	1.0278	1.05	2.000789	3.220307
9	1.0265	1.05	7.704555	8.658747
10	1.0475	1.05	-3.82607	-1.92134
11	1.034867	1.045121	-9.17946	-7.08735
12	1.016341	1.034689	-6.26207	-4.35619
13	0.98355	1.024046	-9.25927	-7.33992
14	0.94799	1.021645	-10.3062	-8.33147
15	0.949097	1.032623	-9.0588	-7.32764
16	0.950643	1.036178	-8.2876	-6.69072
17	0.942051	1.019368	-10.7476	-8.75222
18	0.943435	1.026433	-11.3033	-9.2237
19	1.006535	1.043749	-11.0878	-8.93792
20	0.957431	1.04367	-5.51192	-4.33897
21	0.984316	1.025963	-4.17902	-2.71939
22	1.014535	1.046227	0.35531	1.543027
23	1.011777	1.035025	0.082841	1.324106
24	0.972472	1.010558	-6.64411	-5.01794
25	1.026245	1.050722	-4.82057	-3.23447
26	1.012198	1.047155	-6.0492	-4.40392
27	0.99126	1.032885	-8.16917	-6.38541
28	1.016282	1.046709	-2.3078	-0.86802
29	1.018682	1.047073	0.618242	1.906224
30	0.9866	1.037865	-1.81965	-0.67717
31	0.953749	1.039852	-6.45797	-5.13957
32	0.933722	1.020123	-6.43735	-5.10929
33	0.954825	1.038039	-6.29758	-4.9786
34	0.953278	1.028342	-8.12918	-6.4814
35	0.955743	1.014432	-8.40017	-6.63125
36	0.972852	1.023211	-6.73355	-5.09397
37	0.980415	1.028881	-7.93796	-6.1826
38	0.980119	1.025702	-8.917	-7.05579
39	0.98556	1.040253	-0.8378	0.197616

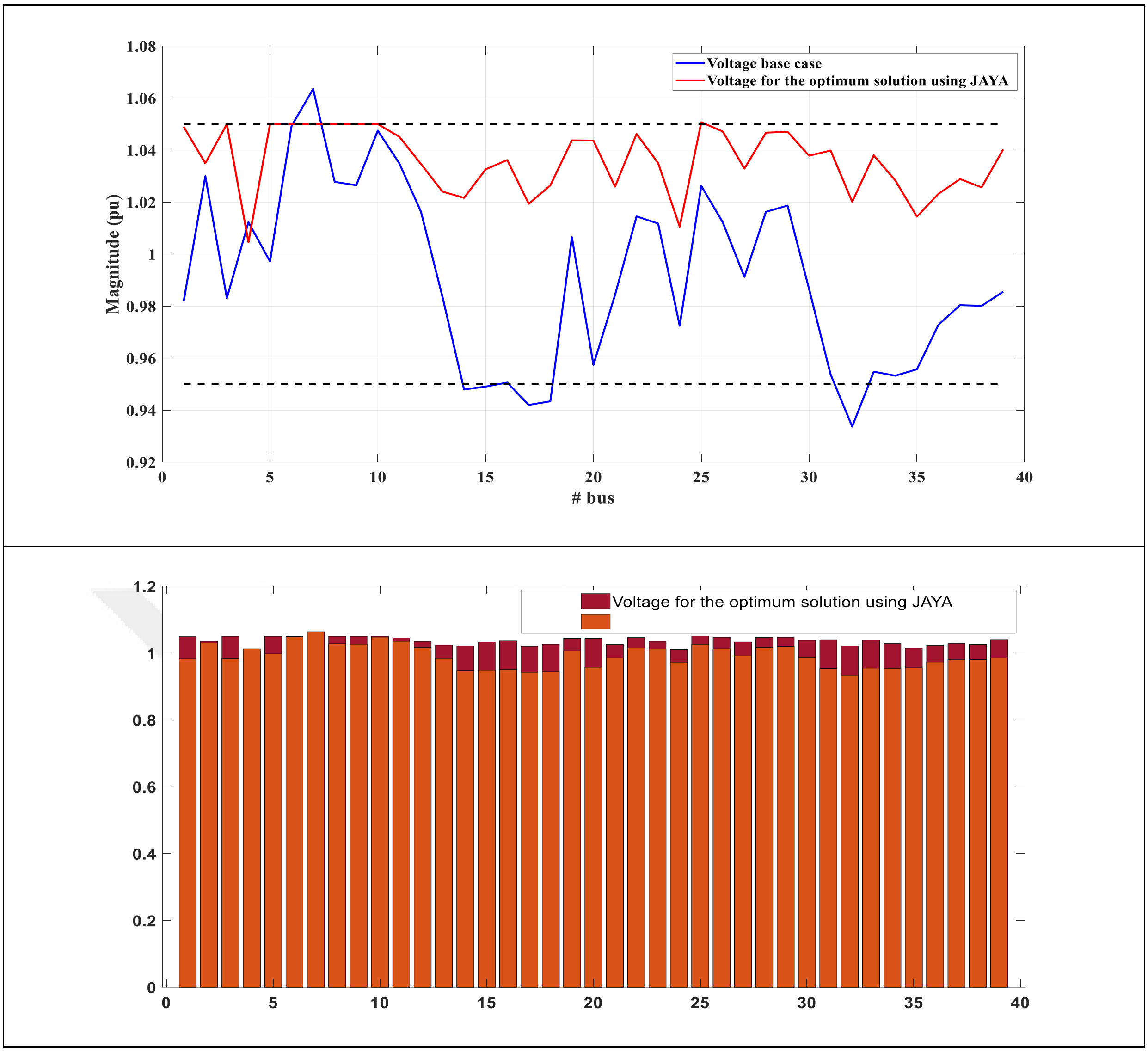

Figure 4.1: Voltage base and voltage optimum with 35 iterations.

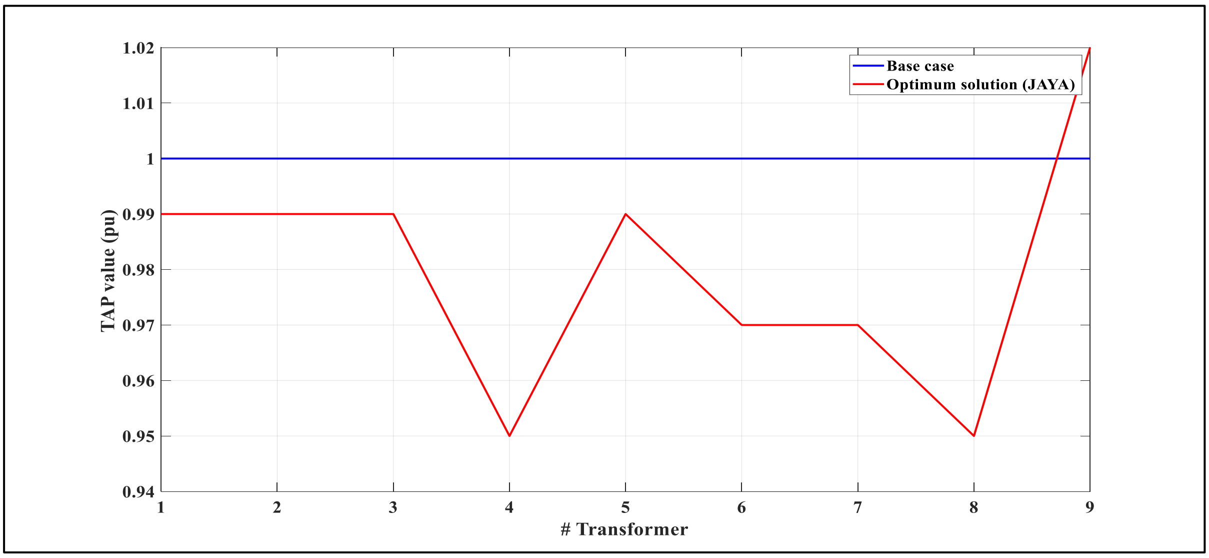

Figure 4.2: Tap-changing transformers.

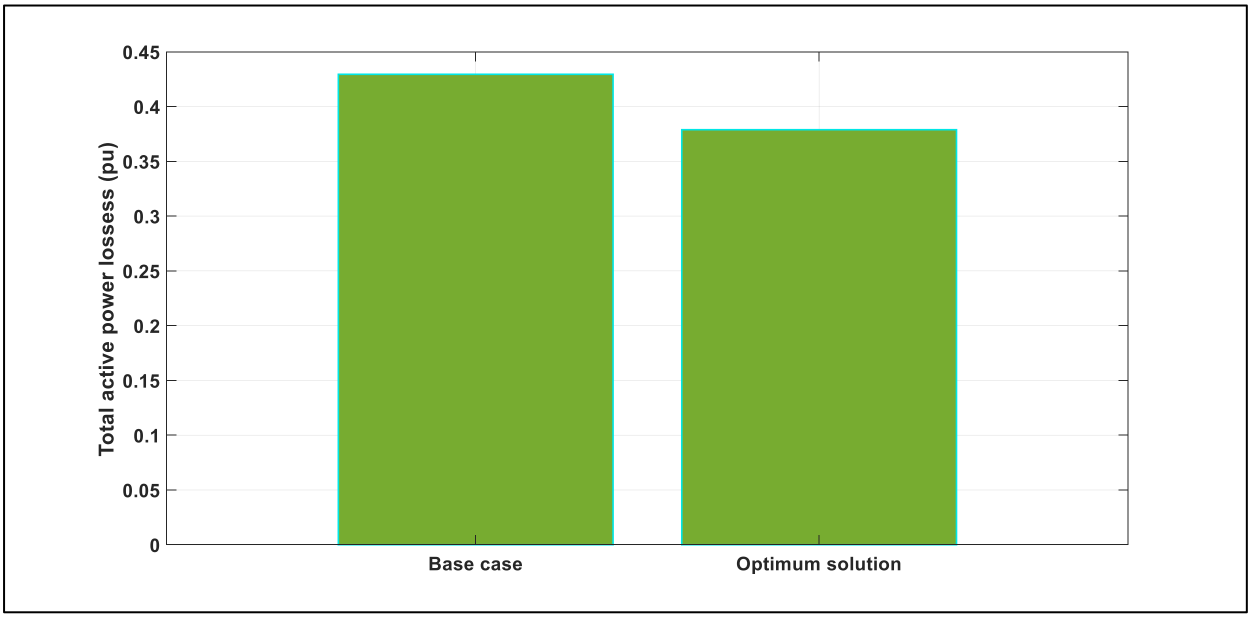

Figure 4.3: Total active power losses for base and optimum solutions.

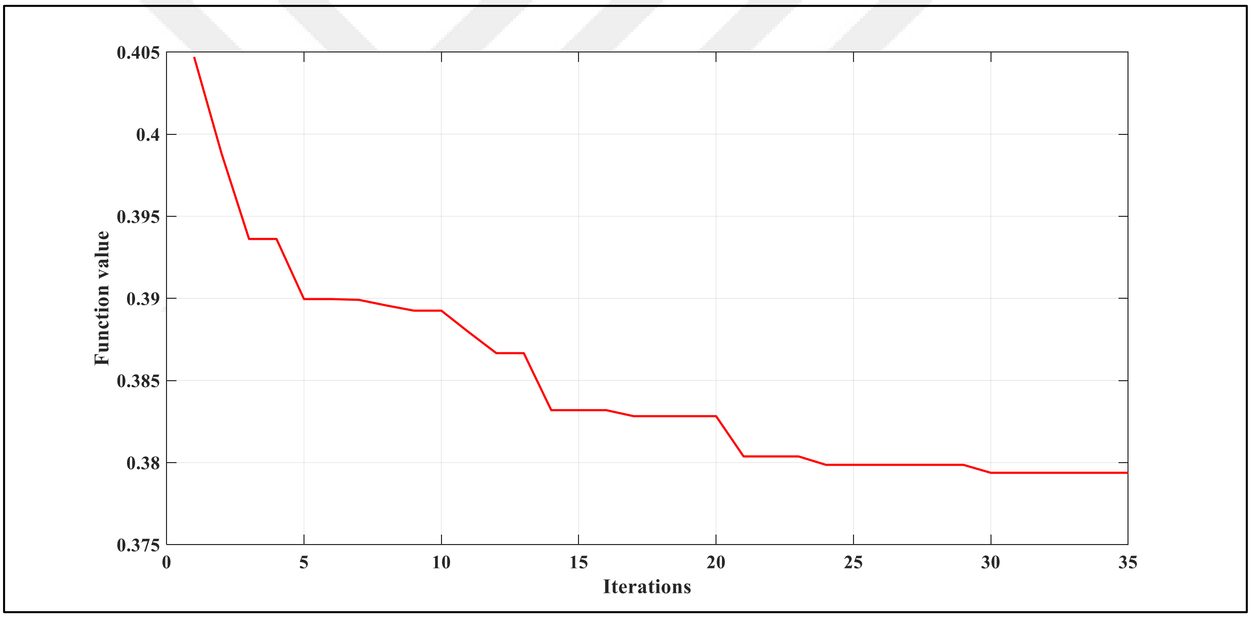

Figure 4.4: Iteration (35).

4.2.1 Active Power Losses Optimal

Jaya optimization has been used to reach the most optimal solution. In this specific scenario, JAYA optimization aims to reduce active power loss. We used the JAYA model to get the best possible solution. The results of this endeavour are presented in Tables 4.1 and 4.2, respectively. Clearly, the JAYA algorithm effectively determines the optimal settings for the system variable, which helps reduce the loss of power wasted by the system. As a direct result, the active power loss is greatly reduced when Jaya algorithm is implemented, from

0.43 pu to 0.33 pu as shown in Figures 4.3 and 4.7. The sharp convergence of the actual power loss can be seen in Figures 4.1 and 4.5, which were generated using the JAYA algorithm.

After 100 iterations, the algorithm reaches the highest level of accuracy, fully obtaining the most optimal solution possible. To assess whether the JAYA algorithm is effective or ineffective, the study results show that the optimal mode for solving the optimal power flow problem has been achieved. The estimated optimal value of the miniaturized active power loss for system optimization is compared with that obtained using the population-based optimization algorithms reported in the literature. Violations of the amount of voltage at each load carrier are shown in Tables 4.1 and 4.2, which explains why the JAYA algorithm is superior to that used in previous studies. These results demonstrate that the JAYA algorithm is unique compared to other methods and techniques described in scientific research in terms of solution improvement and applicability.

4.2.2 Fuel Cost Minimization

During the process of the Jaya algorithm experiment, the goal function that is being looked at is the fuel cost minimization. The convergence graph of the Jaya algorithm's effort to minimize fuel costs is depicted in figure 4.9. The fact that the method requires 100 iterations to obtain the best solution demonstrates how effectively the Jaya algorithm achieves convergence. Tables 4.1 and 4.2 contain a tabular representation of the best possible adjustments that can be made to the design variables, as well as the optimal values for minimizing costs. When the Jaya algorithm is put into action, the results demonstrate a significant reduction in the cost of fuel, which goes from 820.8 $/h to 799.5 $/h. In addition, the typical amount of time spent performing computations for a single iteration of this case is two minutes. These findings demonstrate that the Jaya algorithm is effective in terms of the optimality of the solutions it produces and the speed with which it converges. To further validate the Jaya algorithm, the fuel cost is compared to that obtained using a variety of other empirical optimization algorithms. The JAYA algorithm is superior to any different algorithm that preceded it. Notably, most solutions obtained through practical optimization algorithms are not feasible. This is primarily the magnitude of the voltage resulting from violations in one or more system load buses and violations of power generation ranges in one or more generating units.

As we see in figures 4.3 and 4.7, the standard deviation value computed by the suggested techniques (JAYA) is more significant and higher than that supplied by the actual data, but the total active power loss values derived by the JAYA algorithm are lower. This explains that the proposed improvement approach has achieved optimal results regarding Total Active Power Losses. In addition, the JAYA algorithm often completes its tasks in a far shorter amount of time than the competition. The results show that the JAYA algorithm can provide superior results in much less time than other optimization methods. The fuel cost and active power loss caused by the JAYA algorithm are calculated and explained. The JAYA method returns lower results for all targets than other algorithms mentioned in the literature. The results prove that the proposed JAYA algorithm is the most effective in solving many OPF objectives.

The results are shown in the figures, the models ability to learn fast data and its ability to train quickly, and the low disparity between training data despite the difficulty of learning on such data. This accuracy shows the high efficiency of this model and its ability to deal with such difficult types of data. The results show the superiority of the proposed model over its counterparts from the previous models with increased accuracy, Optimization, and rapid learning on such difficult and convergent data. The exploitation of such a proposed model in the scientific community is very important because it is effective, efficient, and flexible with diverse and different data.

Table 4.2: Original and JAYA results with 100 iterations.

VOLTAGE			ANGLE	
Bus	Orig	JAYA	Orig	JAYA
1	0.982	1.05	0	0
2	1.03	1.034305	-10.8515	-8.63346
3	0.9831	1.05	2.426817	2.27135
4	1.0123	1.05	3.43347	4.018922
5	0.9972	1.05	4.369007	4.899549
6	1.0493	1.029386	5.364398	6.255867
7	1.0635	1.05	8.158688	9.351459
8	1.0278	1.001624	2.000789	3.36514
9	1.0265	1.05	7.704555	8.738728
10	1.0475	1.031301	-3.82607	-1.93556
11	1.034867	1.045243	-9.17946	-7.04722
12	1.016341	1.036071	-6.26207	-4.33843
13	0.98355	1.020328	-9.25927	-7.2982
14	0.94799	1.029523	-10.3062	-8.27735
15	0.949097	1.037324	-9.0588	-7.27994
16	0.950643	1.041261	-8.2876	-6.64931
17	0.942051	1.028149	-10.7476	-8.69038

Table 4.2: Original and JAYA results with 100 iterations "Table Continued".

18	0.943435	1.026679	-11.3033	-9.15961
19	1.006535	1.043448	-11.0878	-8.88291
20	0.957431	1.047482	-5.51192	-4.31217
21	0.984316	1.027231	-4.17902	-2.68112
22	1.014535	1.048107	0.35531	1.561276
23	1.011777	1.037845	0.082841	1.336644
24	0.972472	1.020261	-6.64411	-4.96558
25	1.026245	1.049983	-4.82057	-3.13275
26	1.012198	1.046602	-6.0492	-4.32708
27	0.99126	1.032273	-8.16917	-6.32083
28	1.016282	1.046427	-2.3078	-0.78957
29	1.018682	1.046883	0.618242	1.985513
30	0.9866	1.04579	-1.81965	-0.66203
31	0.953749	1.044107	-6.45797	-5.10859
32	0.933722	1.024422	-6.43735	-5.07855
33	0.954825	1.042224	-6.29758	-4.94893
34	0.953278	1.033365	-8.12918	-6.44294
35	0.955743	1.016395	-8.40017	-6.58254
36	0.972852	1.023811	-6.73355	-5.04409
37	0.980415	1.028225	-7.93796	-6.12985
38	0.980119	1.023871	-8.917	-7.00783
39	0.98556	1.043391	-0.8378	0.216807

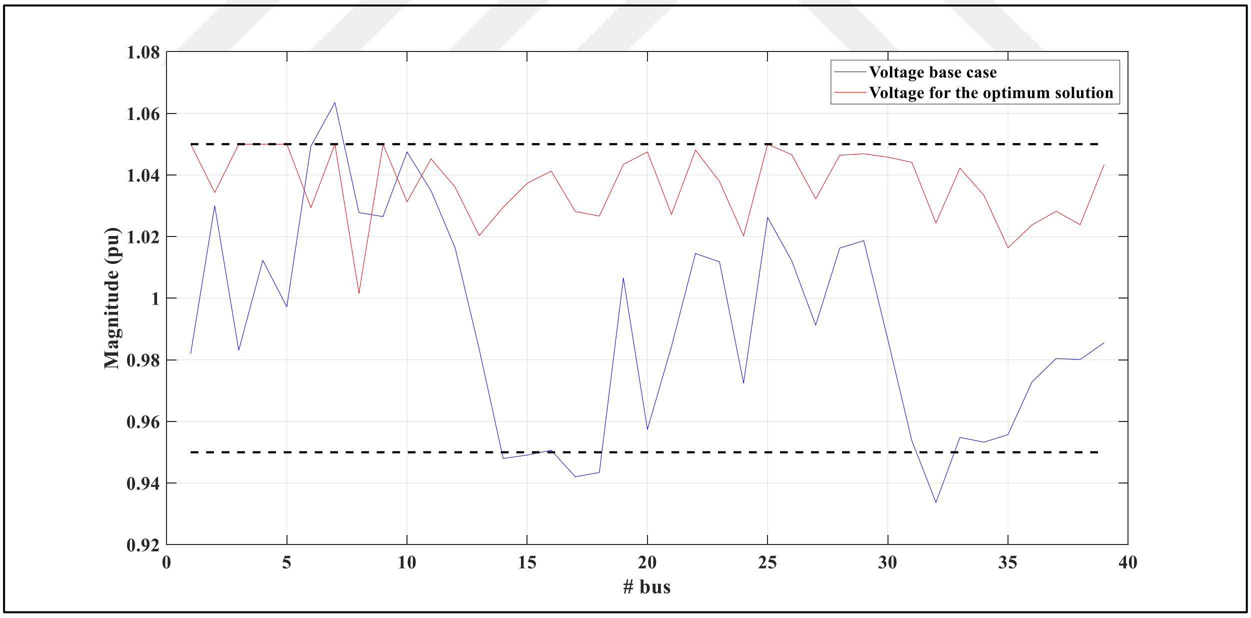

Figure 4.5: Voltage base and voltage optimum 35 iterations.

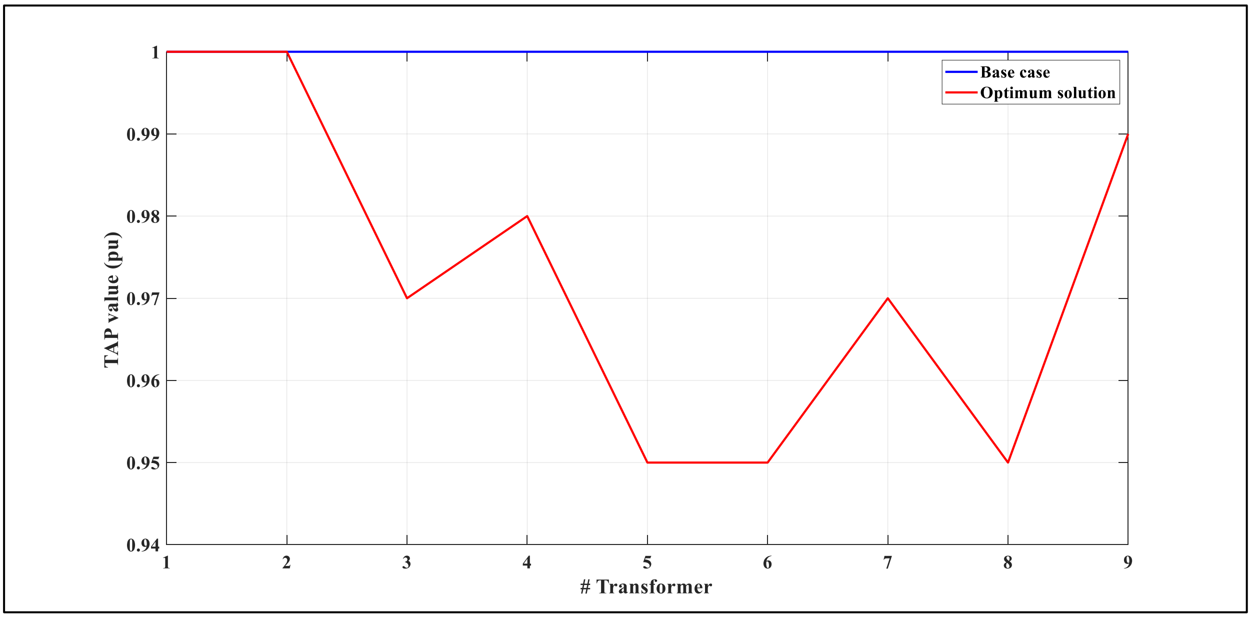

Figure 4.6: Tap-changing transformers.

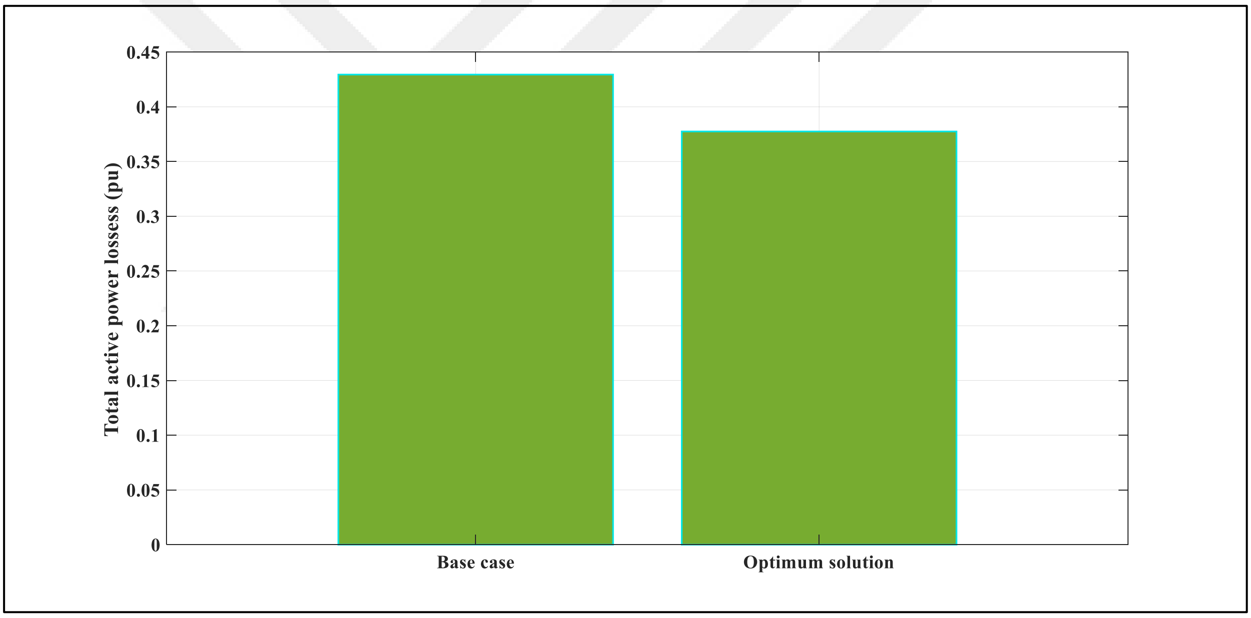

Figure 4.7: Total active power losses for base and optimum solutions.

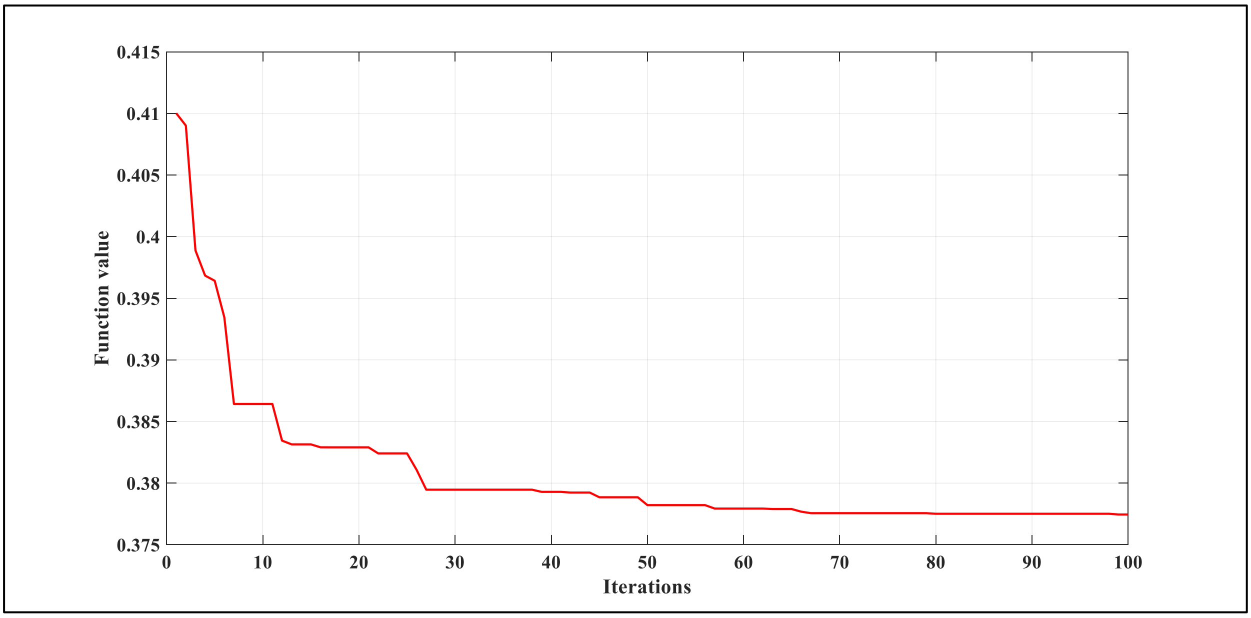

Figure 4.8: Iterations (100).

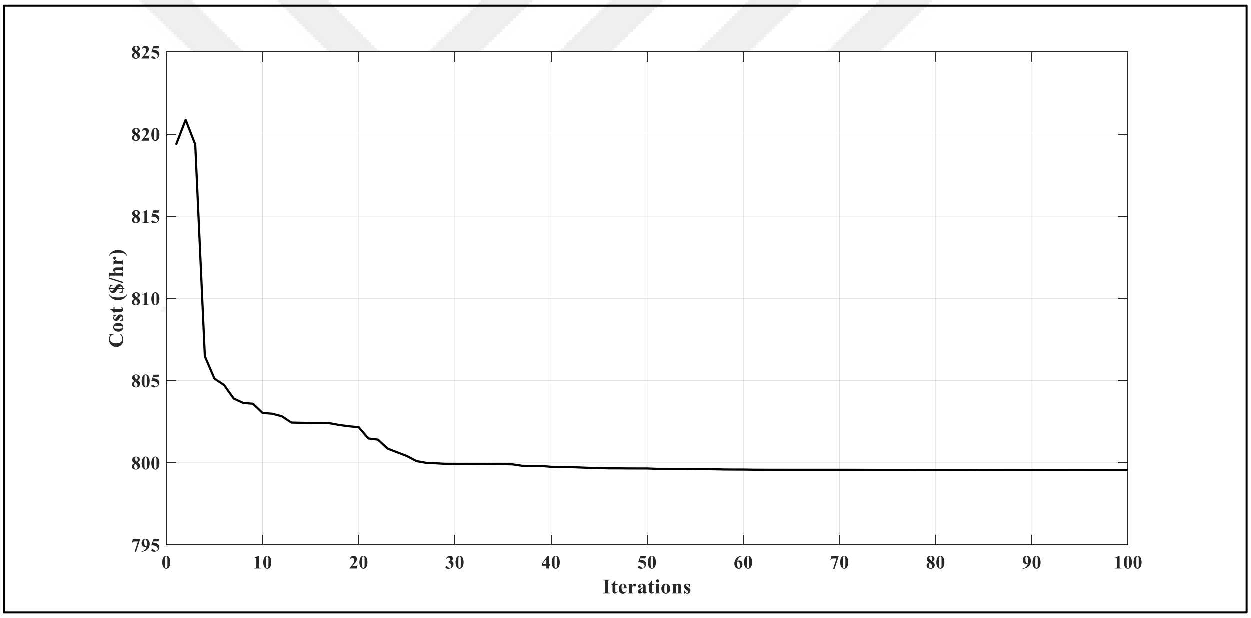

Figure 4.9: Fuel cost minimization with Jaya algorithm.

5 CONCLUSION AND FUTURE WORK

The JAYA algorithm is a powerful yet straightforward global optimization algorithm that can determine which option offers the optimal solution when applied to optimization issues. This algorithm's objective is to establish itself as the preeminent method for arriving at the best possible solution to problems. The JAYA optimization algorithm has many benefits, including the following: The JAYA algorithm requires less time and has fewer parameters than other algorithms. Due to the fact that the JAYA algorithm always comes out on top, it is widely considered to be one of the most effective algorithms. Analytical representation of the JAYA algorithm Calculations can be done to determine the values that have been modified for the JAYA algorithm's control variables.

The problem has been solved for both objective functions in their own right by utilizing various algorithms outlined in the relevant research. JAYA is the name of our proposed algorithm. The IEEE 39-bus system is the one being considered for the test cases. The test cases use data that is taken from the standard for the bus data and the line data. The program utilized is MATLAB, and the number of iterations performed for each of the cases 35 and 100. When the Jaya algorithm is put into action, the results demonstrate a significant reduction in the cost of fuel, which goes from 820.8 $/h to 799.5 $/h. In addition, the typical amount of time spent performing computations for a single iteration of this case is two minutes. These findings demonstrate that the Jaya algorithm is effective in terms of the optimality of the solutions it produces and the speed with which it converges.

The Jaya algorithm is an effective technique for solving problems such as optimization problems in power systems and is a strong competitor for solving OPF problems that arise in practical power systems. It became clear that the JAYA algorithm is superior in terms of optimization and practicality in many studies conducted. JAYA technique is based on a concept that guarantees success that gives it an advantage over other population-based optimization methods in terms of uptime and high accuracy. In addition, the JAYA algorithm approach improves the potential difference and angle, improves generation cost and power loss sensitivities, and finds better solutions. In conclusion, it has been demonstrated that the JAYA algorithm can be applied to solve the problem presented in this thesis and has the potential to be effective in solving OPF problems for both power systems and optimization. In the future, the performance of the JAYA algorithm will be able to compare with that of other global solutions and enable the release of new versions that deal with more complex problems. In addition, it is reliable for real-time analysis of power systems.

REFERENCES

[1] Zhu, Jizhong. Optimization of power system operation. John Wiley & Sons, 2015.

[2] Abido, Mohammad A. "Optimal power flow using particle swarm optimization." International Journal of Electrical Power & Energy Systems 24.7 (2002): 563-571.

[3] Kumar, Sanjeev, and D. K. Chaturvedi. "Optimal power flow solution using fuzzy evolutionary and swarm optimization." International Journal of Electrical Power & Energy Systems 47 (2013): 416-423.

[4] Adaryani, M. Rezaei, and A. Karami. "Artificial bee colony algorithm for solving multi-objective optimal power flow problem." International Journal of Electrical Power & Energy Systems 53 (2013): 219-230.

[5] Dommel, Hermann W., and William F. Tinney. "Optimal power flow solutions." IEEE Transactions on power apparatus and systems 10 (1968): 1866-1876.

[6] Alsac, Ongun, and Brian Stott. "Optimal load flow with steady-state security." IEEE transactions on power apparatus and systems 3 (1974): 745-751.

[7] Sun, David I., et al. "Optimal power flow by Newton approach." IEEE Transactions on Power Apparatus and systems 10 (1984): 2864-2880.

[8] Burchett, R. C., H. H. Happ, and D. R. Vierath. "Quadratically convergent optimal power flow." IEEE Transactions on Power Apparatus and Systems 11 (1984): 3267-3275.

[9] Shoults, Raymond R., and D. T. Sun. "Optimal power flow based upon PQ decomposition." IEEE Transactions on Power Apparatus and Systems 2 (1982): 397-405.

[10] Frank, Stephen, Ingrida Steponavice, and Steffen Rebennack. "Optimal power flow: a bibliographic survey I." Energy systems 3.3 (2012): 221-258.

[11] Lai, Loi Lei, et al. "Improved genetic algorithms for optimal power flow under both normal and contingent operation states." International Journal of Electrical Power & Energy Systems 19.5 (1997): 287-292.

[12] Bakirtzis, Anastasios G., et al. "Optimal power flow by enhanced genetic algorithm." IEEE Transactions on power Systems 17.2 (2002): 229-236.

[13] Kumari, M. Sailaja, and Sydulu Maheswarapu. "Enhanced genetic algorithm based computation technique for multi-objective optimal power flow solution." International Journal of Electrical Power & Energy Systems 32.6 (2010): 736-742.

[14] Abou El Ela, A. A., M. A. Abido, and S. R. Spea. "Optimal power flow using differential evolution algorithm." Electric Power Systems Research 80.7 (2010): 878-885.

[15] Sivasubramani, Swarup, and K. S. Swarup. "Multi-objective harmony search algorithm for optimal power flow problem." International Journal of Electrical Power & Energy Systems 33.3 (2011): 745-752.

[16] He, Xuanhu, et al. "An improved artificial bee colony algorithm and its application to multi-objective optimal power flow." Energies 8.4 (2015): 2412-2437.

[17] Duman, Serhat, et al. "Optimal power flow using gravitational search algorithm." Energy conversion and management 59 (2012): 86-95.

[18] Riva Sanseverino, Eleonora, et al. "Optimal power flow in islanded microgrids using a simple distributed algorithm." Energies 8.10 (2015): 11493-11514.

[19] Bhattacharya, A., and P. K. Chattopadhyay. "Application of biogeography-based optimisation to solve different optimal power flow problems." IET generation, transmission & distribution 5.1 (2011): 70-80.

[20] Christy, A. Ananthi, and P. Ajay D. Vimal Raj. "Adaptive biogeography based predator–prey optimization technique for optimal power flow." International Journal of Electrical Power & Energy Systems 62 (2014): 344-352.

[21] Frank, Stephen, Ingrida Steponavice, and Steffen Rebennack. "Optimal power flow: a bibliographic survey II." Energy systems 3.3 (2012): 259-289.

[22] Rao, R. "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems." International Journal of Industrial Engineering Computations 7.1 (2016): 19-34.

[23] Radosavljević, Jordan, et al. "Optimal power flow using a hybrid optimization algorithm of particle swarm optimization and gravitational search algorithm." Electric Power Components and Systems 43.17 (2015): 1958-1970.

[24] Alsac, O., et al. "Further developments in LP-based optimal power flow." IEEE Transactions on Power Systems 5.3 (1990): 697-711.

[25] Mota-Palomino, R., and V. H. Quintana. "Sparse reactive power scheduling by a penalty function-linear programming technique." IEEE Transactions on Power Systems 1.3 (1986): 31-39.

[26] Burchett, R. C., H. H. Happ, and D. R. Vierath. "Quadratically convergent optimal power flow." IEEE Transactions on Power Apparatus and Systems 11 (1984): 3267-3275.

[27] Dommel, Hermann W., and William F. Tinney. "Optimal power flow solutions." IEEE Transactions on power apparatus and systems 10 (1968): 1866-1876.

[28] Zhang, Y-P., X-J. Tong, and W. U. Felix. "Study on semismooth Newton optimal power flow algorithm based on nonlinear complementarity problem function." PROCEEDINGS-CHINESE SOCIETY OF ELECTRICAL ENGINEERING 24.9 (2004): 130-135.

[29] Cai, Guanglin, Yongjun Zhang, and Zhen Ren. "Optimal power flow algorithm based on nonlinear multiple centrality corrections interior point method." Trans. China Electrotech. Soc 22.12 (2007): 133-139.

[30] Paranjothi, S. R., and K. Anburaja. "Optimal power flow using refined genetic algorithm." Electric Power Components and Systems 30.10 (2002): 1055-1063.

[31] Konak, Abdullah, David W. Coit, and Alice E. Smith. "Multi-objective optimization using genetic algorithms: A tutorial." Reliability engineering & system safety 91.9 (2006): 992-1007.

[32] Hazra, J., and A. K. Sinha. "A multi-objective optimal power flow using particle swarm optimization." European transactions on electrical power 21.1 (2011): 1028-1045.

[33] Abido, Mohammad A. "Optimal power flow using particle swarm optimization." International Journal of Electrical Power & Energy Systems 24.7 (2002): 563-571.

[34] Karaboga, Dervis. An idea based on honey bee swarm for numerical optimization. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.

[35] Karaboga, Dervis, and Bahriye Akay. "A comparative study of artificial bee colony algorithm." Applied mathematics and computation 214.1 (2009): 108-132.

[36] Ayan, Kürşat, and Ulaş Kılıç. "Artificial bee colony algorithm solution for optimal reactive power flow." Applied soft computing 12.5 (2012): 1477-1482.

[37] Rao, R. Srinivasa, 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 1.2 (2008): 116-122.

[38] Özyön, Serdar, et al. "The application of artificial bee colony algorithm for the economic power dispatch with prohibited operating zone." 2012 International Symposium on Innovations in Intelligent Systems and Applications. IEEE, 2012.

[39] Gao, Weifeng, and Sanyang Liu. "Improved artificial bee colony algorithm for global optimization." Information Processing Letters 111.17 (2011): 871-882.

[40] Chen, Tinggui, and Renbin Xiao. "Enhancing artificial bee colony algorithm with self-adaptive searching strategy and artificial immune network operators for global optimization." The Scientific World Journal 2014 (2014).

[41] Rao, R. Venkata, and Hameer Singh Keesari. "Multi-team perturbation guiding Jaya algorithm for optimization of wind farm layout." Applied Soft Computing 71 (2018): 800-815.

[42] Rao, R. Venkata, et al. "An adaptive multi-team perturbation-guiding Jaya algorithm for optimization and its applications." Engineering with Computers 36.1 (2020): 391-419.

[43] Rao, R. Venkata, et al. "Improved multi-objective Jaya optimization algorithm for a solar dish Stirling engine." Journal of Renewable and Sustainable Energy 11.2 (2019): 025903.

[44] Yu, Kunjie, et al. "A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module." Applied Energy 237 (2019): 241-257.

[45] Elattar, Ehab E., and Salah K. ElSayed. "Modified JAYA algorithm for optimal power flow incorporating renewable energy sources considering the cost, emission, power loss and voltage profile improvement." Energy 178 (2019): 598-609.

[46] Chakraborty, Uday K. "Semi-steady-state jaya algorithm for optimization." Applied Sciences 10.15 (2020): 5388.

[47] Mane, S., M. Narsingrao, and V. Patil. "A many-objective Jaya algorithm for many-objective optimization problems." Decision Science Letters 7.4 (2018): 567-582.

[48] Yu, Jiangtao, Chang-Hwan Kim, and Sang-Bong Rhee. "Oppositional Jaya algorithm with distance-adaptive coefficient in solving directional over current relays coordination problem." IEEE Access 7 (2019): 150729-150742.