International Journal of Engineering Research Excellence and Applied Science (IJEREAS)

Volume-II (Issue 3) – September 2024 ISSN: 3048-5355

Comprehensive Analysis of Optimization Techniques for Achieving Optimal Power Flow in Power Systems

Keshav Narayan Meena[1]
M.Tech Scholar

Jitendra Kumar Sharma[2]
Assistant Professor

Dr. Sanjay Kumar Mathur[3]
Principal & Professor

[1,2,3] Department of Electrical Engineering
Aryabhatta College of Engineering and Research Center, Ajmer, Rajasthan, India

Abstract**—** The OPF model (optimal energy flow) is a challenge in calculation of the optimum operating level of an electricity plant, typically to reduce maintenance costs, in order to satisfy the requirements of the whole transmission network. A number of intelligent optimization methods, including non-linear and dynamic optimization problems, have recently been implemented to solve the Optimal Power Flow (OPF) problems. Knowledge optimization approaches rely on several ideas, for example evolutionary heuristic algorithms and a human heuristic algorithm. This article contains the basic information about the optimal power flow and also have discussed the various algorithms such as Cuckoos search algorithm, teaching Learning Algorithm and Genetic algorithm which plays most important role in optimal power flow.

Keywords- *Optimal Power Flow, Teaching Learning Based Optimization, Constraints, Graphical User Interface, IEEE-Bus System.*

Introduction

Modern The OPF model (optimal energy flow) is a challenge in calculation of the optimum operating level of an electricity plant, typically to reduce maintenance costs, in order to satisfy the requirements of the whole transmission network.

Since electricity flows according to the non-linear and non-convex function of the physical characteristics of the system, this can be a problem. However, in real operation, cases with the entire distribution network must be resolved in real time (for many independent network operators, every five minutes) to ensure accurate demand satisfaction.

Better Power Flow is an optimization method for the study, transfer, and power control of the system. The use of optimum power flow is becoming increasingly important due to its ability to manage a variety of scenarios. This issue involves maximizing the objective function, which can take different forms when satisfying a number of practical and physical constraints. The formula for the OPF is presented and various goals and shortcomings are addressed. This article focuses on the stochastic optimization methods used in the literature to solve problems of high power flow. There are also three practical options.

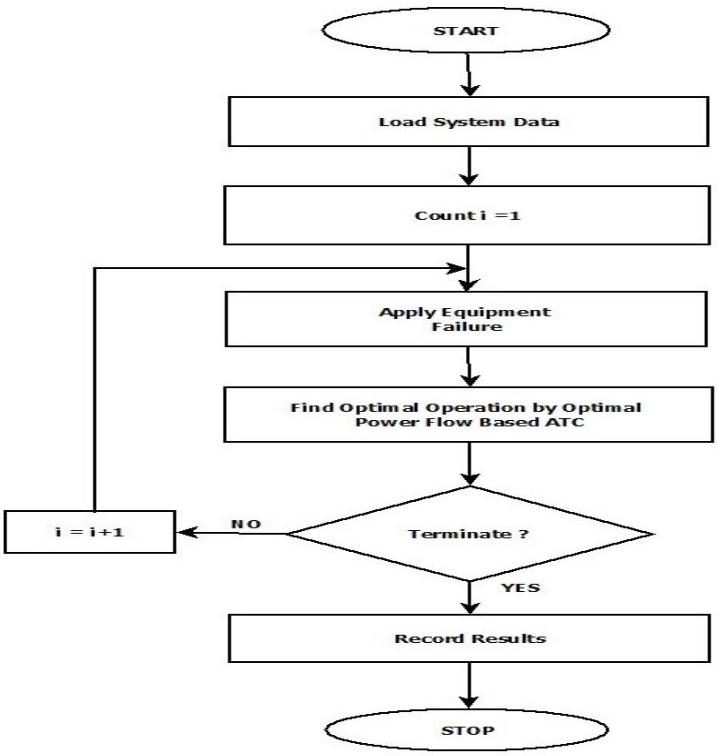

Figure 1 A simple flow chart for optimal power flow

A. Application of optimal power flow

The OPF can help fix a lot of problems. In certain cases, OPF will influence power system resolution [23] in the standard definition of OPF problems, if an empty control set is specified, the algorithm will be automatically reduced to a normal power flow problem. The method in this case relies on the channel mismatch equation and offers the same state-of-the-art solution as traditional energy flow, including channel voltage and bypass flow. OPF can be correlated with a restricted economic task to evaluate the optimum distribution of loads between generators by specifying the characteristics of production costs, network models and load curves. OPF may also be used to mitigate overall loss of active power by reactive transfer of power. In this case, only passive power devices, such as transformer switches, parallel capacitors and reactors, and excitation mechanisms, are used to mitigate the complete failure of the whole grid or sub-network.-The whole network. OPF may be used to define workable alternatives or the so-called minimal control strategy to show if they exist. According to this approach, the goal of the optimization process is to minimize the cost function as a result of the control variance from the reference case [23].

B. Optimal Power Flow Solution Methods

Many traditional methods, including Newton's network flow simulation method and linear programming as well as non-linear programming, square programming and internal points, have been used to address OPF problems. The major drawback of classical methods is that these methods are not suitable for large and complicated problems of nonlinearity and multimodal optimization, so that they can fall to a minimum locally.

A number of intelligent optimization methods, including non-linear and dynamic optimization problems, have recently been implemented to solve the Optimal Power Flow (OPF) problems. Knowledge optimization approaches rely on several ideas, for example evolutionary heuristic algorithms and a human heuristic algorithm.

Figure 2 explains the conventional and modern methods of optimization used to solve the OPF problem

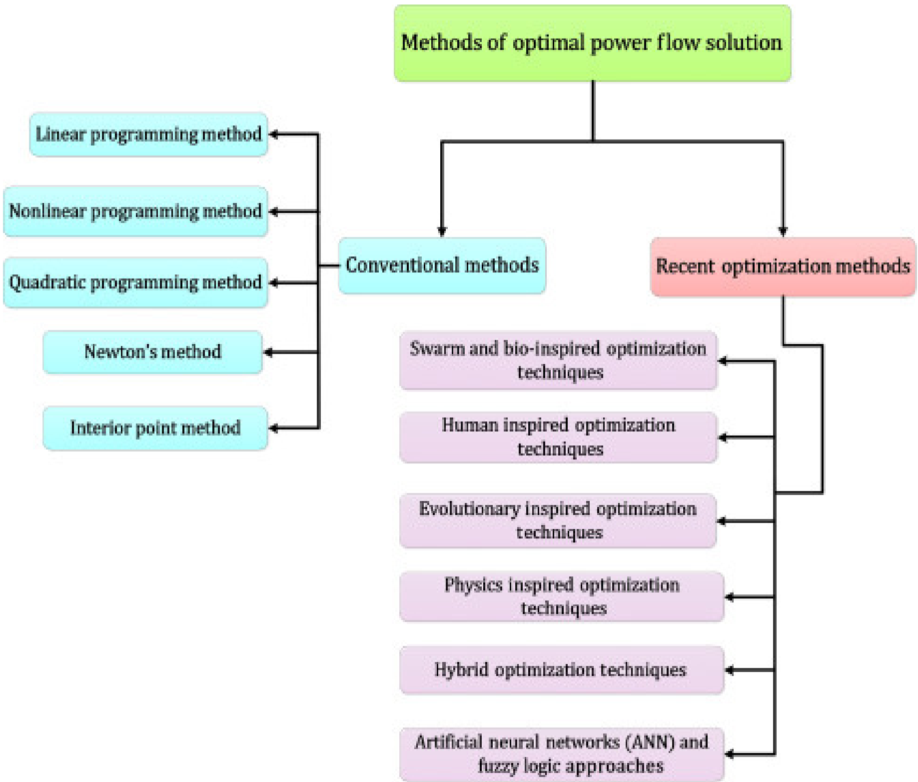

Figure 2 Optimal power flow solution methods

OPF problem Optimization Methods

A. Cuckoos search algorithm

Check Cuckoo is one of the existing heuristic algorithms widely used in different fields of engineering to address optimization problems. The method to solve globally optimized problems is extremely efficient since it can use toggle parameters to balance local and global random cycles. The parameter for the toggle of the original Cuckoo Search Algorithm is fixed at 25%, and the effect on performance of the Cuckoo Search Algorithm has not been examined.

In this article three new Cuckoo search algorithms are built based on a dynamically enhanced conversion parameter. The results are compared to results of the search algorithm Cuckoo, each with set and dynamically reduced swapping parameters. This search is carried out in the form of ten Cuckoo search algorithms. Finally, the findings of the simulation analysis reveal that the cuckoo search algorithm is stronger than other cuckoo search algorithm, with the swapping parameters increasing exponentially.

2. A usual heuristic algorithm based on the raising of cuckoo birds, the Cuckoo Search Algorithm [14] is used. It is really appropriate to use the CS algorithm to combine possible solutions with the cuckoo eggs. In the nests of other cuckoos Rhododendrons usually place their fertilized eggs, assured of their adoptive parents to pick up their offspring.

The cucumbers cannot find that the eggs are in their nest. In this case, the exterior eggs are either killed or cast out of the entire nest. In general, the CS optimization algorithm is based on three rules:

3. Each cuckoo randomly chooses a nest and lays eggs.

4. The best high quality egg nest will be presented to the next generation.

5. For a specified number of nests, the probability that the cuckoo will find sibling eggs is pε [0.1]. In this case, the cuckoo can lay the eggs or lay the nest and then make the nest elsewhere.

A new random solution can be used to replace the decimal pa of the host nest with a new nest. Clearly, the coherence or operation of the solution is proportional to its value. The recommendation is: each egg in the nest is a solution, each cuckoo may lay only one Egg (thus a solution). The following description is given. We may differentiate chickens, nests or cuckoos without discrimination. The aim is to substitute a new and probably better solution for the wrong solution in the nest (cuckoo egg).

The basic steps of a three rule cuckoo finder algorithm can be summarized like the pseudo code shown in Figure 3.

Objective function $f(x)$, $x = (x_1, x_2 \dots \dots x_d)^T$
Generate initial population of n host nests xi ($i = 1, 2 \dots n$)
While (t < Max Generation) or (stop criteria)
Get a cuckoo (say i) randomly by Lévy distribution;
Evaluate its quality/fitness F_i;
Choose a nest among n (say j) randomly;
Evaluate its quality/fitness F_j;
If ($F_i > F_j$)
Replace j by the new solution;
End
A fraction of (P_a) of worse nests are abandoned and
new ones are built at new locations via Lévy flights;
Keep the best solutions (or nests with quality solutions);
Rank the solutions and find the current best;
End while
Post processing

Figure 3 Pseudo code

CS is one of Yang and Deb's most recent natural algorithms. The parasitism of some rhododendron species is the subject of the CS. Moreover, the so-called Lévy flight multiplies the algorithm and not just an isotropic random path. According to recent research, CS is effective rather than PSO and GA.

In addition, the cuckoo has unbelievable abilities in nesting other host birds (usually of other species), such as the discovery of a nest with freshly laid eggs and the removal of existing eggs to increase the incubation time of eggs. Any host bird can resist the cuckoo's parasite action and lay foreign eggs found at various places or construct nests. For CS algorithm growth, this proportion of breeding cuckoos is used. Physical systems are so complex that computer algorithms can not be interpreted in their straightforward form. Physical structure simplification is essential for the effective use of computer algorithms. The cuckoo reproduction system was also simplified by Yang and Deb into three idealized regulations. (1) The egg is a nest keeping solution. Just one egg can be laid by artificial cuckoos. (2) In order to boost the survival rate of his larvae, the cuckoo would find the most suitable nest for laying eggs (solution). In order to ensure that only quality eggs which look better like host birds (the best choice at the best price) have the potential to evolve (next generation) and become mature cuckoos (3) are introduced an elite breeding policy. There are clear percentages of host nests (populations). The host bird will be faced with the discovery of foreign eggs (the best advantage for poor decline), the dumping of eggs or the abandonment of the nest, and the building of a whole new nest in a new location. Otherwise, in the next generation the eggs will mature and survive. The new eggs laid by the cuckoo have chosen the Lévy flights to build a nest around the best solution possible.

In CS mode, from the point of departure to the duplication of the full number of genes, the population of egg (atom) grows. Each egg is a dimension vector, which corresponds with each part to the decision variable of the optimization problem to be discussed. Each egg (candidate solution) is measured according to an objective method and the value of its suitability is the final product. The evolutionary process of CS is represented by three different operators: (a) Lan Fei, (b) the replacement of new solutions and (c) the technique of elite selection for nest generation.

B. Teaching Learning Algorithm

Swarm creation and intelligence algorithms, which have general control parameters (for example population size , number of generations, and elite), are probabilistic algorithms. Besides general parameters of control, various algorithms need their own unique algorithm control parameters. For example, GA uses mutation chances, crossing chances and the selection operator. PSO uses mental and inertial weights mentally. Spectators, licensed bees, identification bees are used by ABC, and the number of bees are reduced. The HS algorithm uses the variable, phase changing variable and number of improvisations for harmony memory evaluation. Likewise, other algorithms (such as ES, EP, DE, SFL, ACO, FF, CSO, AIA, GSA, BBO, FPA, ALO, IWO, etc.) need to adjust their parameters for the algorithm. Correct adjustment of specific algorithm parameters is a very critical factor, which will affect the performance of the above algorithm. Incorrect modification of the parameters of this algorithm would increase the size of the calculation. Rao et al, because of this fact. An optimization was introduced by the Teaching and Learning Algorithm (TLBO) (2011,) which does not require parameters for such algorithms. The TLBO algorithm only includes general regulatory parameters (e.g. population size, generations). The TLBO algorithm has now been generally recognized by maximizing experts.

The work of TLBO is divided into two components: 'teacher level' and.' below are the functions of these two measures.

1. Teacher phase

This is the first part of the algorithm that the student learns through the teacher. At this stage, the teacher will try to improve the average grade of the courses he teaches based on his abilities. In each iteration i, assume that there are "m" topics (eg: design variables), "n" students (i.e. population size, k = 1,2, ..., n) and Mj, i is the mean result of the specific topic "The ratio of students to" j "(j = 1,2, ..., m) is the best overall result Xtotal-kbest, I consider the sum of all topics in the whole group of students and can be considered However, because teachers are generally considered to be highly educated individuals who can train students to achieve better grades, the algorithm considers them to be the best learners identified as teachers. each lesson and the corresponding teacher, the result for each lesson is:

$$Difference\_Mean_{j,k,i} = r_i\,(X_{j,kbest,i} - T_F M_{j,i}) \qquad (1)$$

Where, $X_{j,kbest,i}$ is the result of the best learner in subject j. T_F is the teaching factor which decides the value of mean to be changed, and r_i is the random number in the range [0, 1]. Value of T_F can be either 1 or 2. The value of T_F is decided randomly with equal probability as,

$$T_F = round\,[1+rand\,(0,1)\{2\text{-}1\}] \qquad (2)$$

T_F is not a parameter of the TLBO algorithm. The value of T_F is not given as an input to the algorithm, and its value is randomly determined by the algorithm using an equation. (2). After a lot of experiments on many benchmark functions, the conclusion is that if the value of T_F is between 1 and 2, the performance of the algorithm is better. However, if the value of T_F is 1 or 2, the performance of the algorithm is much better, and therefore, to simplify the algorithm, it is recommended to use 1 or 2 as the teaching factor according to the rounding standard given by the equation. (2). Based on Difference_Meanj, k, i, update the existing solution according to the following expression at the teacher stage.

$$X'_{j,k,i} = X_{j,k,i} + Difference\_Mean_{j,k,i} \quad (3)$$

Among them, $X'_{j,k,i}$ are the updated values of $X'_{j,k,i}$. If $X'_{j,k,i}$ provide better function values, it is acceptable. At the end of the teacher phase, all acceptable operating values will remain unchanged and these values will become the entry of the student phase. The student stage depends on the teacher stage.

2. Learner phase

This is the second component of the algorithm of which the students are more sensitive. A student inherently communicates with other students to develop their skills. If a subject has more knowledge than the other subject, more knowledge will be found. According to the population dimension of 'n,' the dynamics of researching this step are discussed below.

Randomly select two learners P and Q such That $X'_{total-P,i} \neq X'_{total-Q,i}$ (where, $X'_{total-P,i}$ and $X'_{total-Q,i}$ are the updated function values of $X_{total-P,i}$ and $X_{total-Q,i}$ of P and Q respectively at the end of teacher phase)

$$X''_{j,P,i} = X'_{j,P,i} + r_i\,(X'_{j,P,i} - X'_{j,Q,i}),\ \text{If } X'_{total-P,i} < X'_{total-Q,i} \quad (4)$$

$$X''_{j,P,i} = X'_{j,P,i} + r_i\,(X'_{j,Q,i} - X'_{j,P,i}),\ \text{If } X'_{total-Q,I} < X'_{total-P,i} \quad (5)$$

$X''_{j,P,I}$ is accepted if it gives a better function value.

The Eqs. (4) and (5) are for minimization problems. In the case of maximization problems, the Eqs. (6) and (7) are used.

$$X''_{j,P,i} = X'_{j,P,i} + r_i\,(X'_{j,P,i} - X'_{j,Q,i}),\ \text{If } X'_{total-Q,i} < X'_{total-P,i} \quad (6)$$

$$X''_{j,P,i} = X'_{j,P,i} + r_i\,(X'_{j,Q,i} - X'_{j,P,i}),\ \text{If } X'_{total-P,i} < X'_{total-Q,i} \quad (7)$$

C. Genetic algorithm (Hereditary Algorithm)

The genetic algorithm represents a random search algorithm aimed at imitating the selection process of the natural genetics. Genetic algorithm is based on string structures (e.g. biological structures), which shift over time by spontaneous yet organized transfers of knowledge based on the fittest 's survival. The most important portion of the old set is also used for generating a new set of strings with each development. The principal characteristics of the genetic algorithm are:

(1) The genetic algorithm encodes the set of parameters instead of the parameter itself.

(2) The genetic algorithm starts the search from a group of points instead of a single point.

(3) The genetic algorithm uses income information instead of derivatives.

(4) Genetic algorithms use probabilistic transition rules instead of deterministic rules.

Next, it must be decided the encoding to be used. Then build the initial string using a random procedure. Then use a group of operators to establish the original population, in order to develop these populations over time. Replication, crossing and mutation are the key operators in genetic algorithms.

In genetic algorithms there are two computing operations: genetic and evolutionary. In order to pick some successful chromosomes to replicate with certain probability, genetic treatment employs the concept of probability transfer when other secondary chromosomes die. The research advice will also be targeted to the most promising area. They will concurrently investigate multiple search area by using random search algorithms, so that it is impossible that the search will stop at least locally. The benefit of the genetic algorithm is that it enables discovery in a comparatively small computational period in various areas in the search field. Many complicated lenses can also be used easily. However, the genetic algorithm offers only a basic structure for addressing complex problems of optimization. Generally, geneticists are connected to challenges and are vital to addressing realistic issues effectively. In particular, a procedure must be used to calculate the resulting heat load in the superstructure for the overall issue of the network of heat exchangers and multi-flux heat exchangers. Some operators, including the crossover operator, the mutation operator, the rectangular and the real accumulation operator, have been designed to deal adequately with the problem of composition. The handling of restrictions is another challenge in implementing genetic algorithms. In the genetic evolution process, after manipulating genetic manipulators, an person in the population can become an impossible solution that will lead to the impossible solution during the course of growth, particularly to optimization challenges with severe restrictions. Consequently, some genetic restriction strategies should be created.

RESULTS & discussion

OPF remains a critical task in modern power systems, ensuring economic and reliable operation while adhering to operational constraints. This discussion highlights the significance of the studied algorithms and their comparative strengths in achieving optimal solutions. Among the optimization techniques explored, the Cuckoo Search Algorithm (CSA) stands out for its simplicity and effectiveness in handling nonlinear and non-convex optimization problems. Its bio-inspired approach, based on the brood parasitism of cuckoo birds, makes it particularly well-suited for exploring and exploiting the solution space in OPF problems. The Teaching-Learning-Based Optimization (TLBO) algorithm offers a unique pedagogical perspective, mimicking the interaction between teachers and students to improve solutions iteratively. Its parameter-less nature enhances its adaptability and reduces computational complexity, making it a reliable tool for achieving OPF solutions. The Genetic Algorithm (GA), a classic evolutionary optimization technique, continues to demonstrate robust performance in OPF applications. Its ability to work with diverse populations and perform crossover and mutation enables it to avoid local optima, making it a versatile choice for solving complex power system problems. Each of these algorithms brings distinct advantages to the OPF landscape, and their performance varies depending on the nature of the power system under study, including its size, topology, and constraints. This article underscores the need for selecting optimization techniques based on specific system characteristics and requirements, often necessitating hybrid approaches or algorithm customization to achieve superior results.

REFERENCES

1. O.Alsac, and B. Scott, "Optimal load flow with steady state security", IEEE Transaction PAS -1973, pp. 745-751. May 1973.
2. R.Narmatha Banu, D.Devaraj "Enhanced Genetic Algorithm Approach for Security Constrained Optimal Power Flow Including FACTS Devices", International Journal of Electrical and Electronics Engineering, pp. 552-557, 2009
3. P.Somasundaram ,K.Kuppusamy , and Kumudini Devi, "Evolutionary programming based security constrained optimal power flow", Electric Power System Research 72, Elsevier pp. 137-145 ,2004
4. H. W. Dommel and W. F. Tinney, "Optimal Power Flow Solutions", IEEE Transactions on Power Apparatus and Systems, Vol. pp. 1866-1876, Oct. 1968.
5. Anastasios G. Bakirtzis, Pandel N. Biskas, Christoforos E. Zoumas and Vasilios Petridis, "Optimal Power Flow by Enhanced Genetic Algorithm" IEEE transactions on Power Systems, Vol. 17, No. 2, pp. 229-236, May 2002.
6. A. Monticelli , M .V.F Pereira ,and S. Granville , "Security constrained optimal power flow with post contingency corrective rescheduling" , IEEE Transactions on Power Systems: PWRS-2, No.1, pp. 175-182, 1987.
7. Pmg Yan and Arun Sekar, "A new approach to security constrained optimal power flows", Power Engineering Society Summer Meeting, IEEE Volume: 3, pp. 1462-1467, 2001.
8. Florin Capitanescu and Louis Wehenkel, "Improving the Statement of the Corrective Security-Constrained Optimal Power-Flow Problem", IEEE Transactions on Power Systems, Vol. 22, No. 2, pp. 887-889, May 2007.

9. Florin Capitanescu and Louis Wehenkel, "A New Iterative Approach to the Corrective Security-Constrained Optimal Power Flow Problem", IEEE Transactions on Power Systems, Vol. 23, No. 4, pp. 1533-1541, Nov 2008.
10. A. J. Wood and B. F. Wollenberg, "Power Generation Operation and Control", New York, NY: John Wiley & Sons, Inc., 1996.
11. Kalyanmoy Deb, "Multi-Objective Optimization using Evolutionary Algorithms", John Wiley& Sons, 2001.
12. C.L. Wadhwa, "Electrical Power Systems", New Age International Publishers, 2009.
13. W. Stagg Glenn, H. El Abiad Ahmed, "Computer Methods in Power System Analysis," McGraw Hill international Book Company, 1968.
14. Hans Glavitsch, Rainer Bacher, "Optimal Power Flow Algorithms" Swiss Federal Institute of Technology CH-8092 Zurich, Switzerland.
15. Amrane, Y.; Boudour, M.; Ladjici, A.A.; Elmaouhab, A. Optimal VAR control for real power loss minimization using differential evolution algorithm. Int. J. Electr. Power Energy Syst. 2015, 66, 262–271.
16. Biswas, P.P.; Suganthan, P.N.; Mallipeddi, R.; Amaratunga, G.A.J. Optimal power flow solutions using differential evolution algorithm integrated with effective constraint handling techniques. Eng. Appl. Artif. Intell. 2018, 68, 81–100.
17. Elsaiah, S.; Cai, N.; Benidris, M.; Mitra, J. Fast economic power dispatch method for power system planning studies. IET Gener. Transm. Distrib. 2015, 9, 417–426.
18. Bhowmik, A.R.; Chakraborty, A.K. Solution of optimal power flow using nondominated sorting multi objective gravitational search algorithm. Int. J. Electr. Power Energy Syst. 2014, 62, 323–334.
19. Chaib, A.E.; Bouchekara, H.R.E.H.; Mehasni, R.; Abido, M.A. Optimal power flow with emission and non-smooth cost functions using backtracking search optimization algorithm. Int. J. Electr. Power Energy Syst. 2016, 81, 64–77.
20. Warid, W.; Hizam, H.; Mariun, N.; Abdul-Wahab, N. Optimal Power Flow Using the Jaya Algorithm. Energies 2016, 9, 678.

21. He, X.; Wang, W.; Jiang, J.; Xu, L. An Improved Artificial Bee Colony Algorithm and Its Application to Multi-Objective Optimal Power Flow. Energies 2015, 8, 2412–2437.
22. Niknam, T.; Narimani, M.R.; Jabbari, M.; Malekpour, A.R. A modified shuffle frog leaping algorithm for multi-objective optimal power flow. Energy 2011, 36, 6420–6432.
23. Sahu, S.; Barisal, A.K.; Kaudi, A. Multi-objective optimal power flow with DG placement using TLBO and MIPSO: A comparative study. Energy Procedia 2017, 117, 236–243.
24. Kim, H.-Y.; Kim, M.-K.; Kim, S. Multi-Objective Scheduling Optimization Based on a Modified Non-Dominated Sorting Genetic Algorithm-II in Voltage Source Converter−Multi-Terminal High Voltage DC Grid-Connected Offshore Wind Farms with Battery Energy Storage Systems. Energies 2017, 10, 986.
25. Zhang, J.; Tang, Q.; Li, P.; Deng, D.; Chen, Y. A modified MOEA/D approach to the solution of multi-objective optimal power flow problem. Appl. Soft Comput. 2016, 47, 494–514.
26. Yu, W.-J.; Ji, J.-Y.; Gong, Y.-J.; Yang, Q.; Zhang, J. A tri-objective differential evolution approach for multimodal optimization. Inf. Sci. 2018, 423, 1–23.
27. Barocio, E.; Regalado, J.; Cuevas, E.; Uribe, F.; Zúñiga, P.; Torres, P.J.R. Modified bio-inspired optimization algorithm with a centroid decision making approach for solving a multi-objective optimal power flow problem. IET Gener. Transm. Distrib. 2017, 11, 1012–1022.
28. Bhowmik, A.R.; Chakraborty, A.K.; Babu, K.N. Multi objective optimal power flow using NSMOGSA. In Proceedings of the 2014 International Conference on Circuits, Power and Computing Technologies [ICCPCT], Nagercoil, India, 20–21 March 2014; pp. 84–88.