



ijerr

Vol 04 issue 10
Category: Research
Received on:29/03/12
Revised on:12/04/12
Accepted on:24/04/12

REACTIVE POWER OPTIMIZATION USING SOFT COMPUTING TECHNIQUES - A COMPARATIVE STUDY

K. S. Chandragupta Mauryan¹, K. Thanushkodi², A.Sakthisuganya¹

¹Department of Electrical and Electronics, Sri Krishna College of Technology, Coimbatore, Tamilnadu

²Akshaya college of Engineering and Technology, Coimbatore, Tamilnadu

E-mail of Corresponding Author: sakthisugan.a@gmail.com

ABSTRACT

In this paper we have analyzed the various algorithms used for reactive power optimization (RPO). The objective of an optimal power flow (OPF) algorithm is to find the steady state operation point which reduces the generation cost, loss etc. Reactive power supply plays an important role in the power system. A properly designed optimal power flow solution provides the best and the most optimum practical solution to achieve improvement in a single or multiple hierarchical objectives based on the various constraints on the system. The main aim of RPO is to minimize the system losses and to maintain the voltage profile according to the constraints. Considering the voltage security, power systems are provided with a lot of voltage controlling devices such as generators, tap changing transformers, shunt capacitors/reactors, synchronous condensers, and static VAR compensators etc. A real time control employing those controlling devices is required to solve the problems. Traditionally, classical optimization methods were used to solve this problem, but due to its various limitations, modern methods have been proposed in this paper for the optimization of the reactive power. The purpose of this paper is to present a comprehensive survey of various reactive power optimization methods.

Keywords: Reactive Power Optimization, Optimal Power Flow, Algorithm, Optimal solution, Global Optimum.

INTRODUCTION

Power flow studies are important for effective planning and operation of power systems as well as in determining the best operation of other existing systems. The main information obtained from the power flow study is the magnitude and phase angle of the voltage at each bus, and the real and reactive power flowing in each line for specified load and generator real power and voltage conditions [1]. After obtaining this information, real and reactive power flow on

each branch as well as generator reactive power output can be determined analytically. The classical methods for solving RPO problem have various limitations and the convergence value move away from local optima. Hence few techniques have been proposed for solving RPO problem effectively. In this paper, a review of various optimization methods for RPO has been presented: (1) Gradient Search Method (2) Differential Evolution Algorithm (3) Biogeography-Based Optimization (4) Tabu Search Algorithm (5) Modified Shuffled Frog

Leaping Algorithm (6) Particle Swarm Optimization (7) Harmony Search Algorithm (8) Artificial Bee Colony Algorithm (9) Ant colony optimization (10) Gravitational Search Algorithm.

REACTIVE POWER OPTIMIZATION

The reactive power optimization problem is one of the most important aspects in economic operation of power system. It is a multi-constraint, multi-modal, mixed-variable, and nonlinear planning problem. It involves the efficient use of all available reactive power sources and control of voltages at voltage-controlled buses so as to minimize the total network active power loss with the satisfaction of all operation constraints [2]. Such operation constraints may include the capacity of reactive power sources, limit of bus voltages and transformer tap position. On account of reducing system losses, appreciable MW savings, or in other words, cost savings can be achieved. Because of this goal, the reactive power dispatch problem has been formulated as a complicated constrained optimization problem with partially discrete, partially continuous and non-differentiable nonlinear objective function. The primary objective of RPO problem is to reduce system power losses and to obtain the setting of various controls, for the same. The real power loss is a non-linear function of bus voltage magnitudes and phase angles, which are implicitly a function of the control variables. Though reactive power does not have its own production cost, it influences the overall production cost as it involves the transmission losses. Hence the main motivation of RPO is to reduce the active power losses in the transmission network [3].

$$\text{Min}P_L = \sum G_K [V_i^2 + V_j^2 - 2V_iV_j \cos(\alpha_i - \alpha_j)] \quad (1)$$

The number of lines in the given network is denoted by K. The minimization of the above function is subjected to a number of constraints.

$$0 = P_{gi} - P_{di} - V_i \sum V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) \quad (2)$$

$$0 = Q_{gi} - Q_{di} - V_i \sum V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) \quad (3)$$

$$\begin{aligned} Q_{ci} \min < Q_{ci} < Q_{ci} \max_{i \in n_c} \\ Q_{gi} \min < Q_{ci} < Q_{ci} \max_{i \in n_g} \\ T_k \min < T_k < T_k \max_{k \in n_t} \\ V_i \min < V_i < V_i \max_{i \in n} \end{aligned} \quad (4)$$

Here n is the total number of buses, n_g and n_c are number of generator and reactive sources, n_t are the number of tap changers. Power flow equations (2) and (3) are used as equality constraints, reactive power source installation restrictions, reactive generation restrictions, transformer tap-setting restrictions, bus voltage restrictions (4) are used as inequality constraints.

DEVELOPMENTS OF RPO

ALGORITHMS

A. GRADIENT METHOD

The gradient method is applied to the RPO problem with the idea of eliminating the existence of the concept of the state and control variables, with load flow equations providing a nodal basis for the elimination of state variables. This helps in obtaining a reduced problem in the space of the control variables with the load flow equations [4]. Salient features are:

- Penalty function optimization approach is used to develop nonlinear programming (NLP) method for minimization of fuel cost and active power losses.
- It also involves verification of boundary, using Lagrange multiplier approach, is achieved. Capable of solving large size power system problems up to 500 buses.

- Its drawback is in the modeling of components such as transformer taps that are accounted in the load flow but not in the optimization routine.

B. DIFFERENTIAL EVOLUTION

Differential evolution is a simple direct search optimization method used to minimize nonlinear and non-differentiable continuous space functions with real-valued parameters. It has been extended to handle mixed integer discrete continuous RPO problem [5]. Design principles are:

- It has a simple structure, easy for usage and known for its robustness.
- Effective for integer, discrete and mixed parameter optimization. Handling non-differentiable, noisy and or time dependent objective functions.
- Effective for nonlinear constraint optimization problems with penalty function [6].

It has a population of candidate solution called agents. An initial population composed of vectors u_{0i} , $i=1, 2, np$, is randomly generated within the parameter space. The adaptive scheme used ensures that the mutation increments are automatically scaled to the correct magnitude. The steps involved in this algorithm are described as follows:

- The agents are initialized as x with random positions in the search-space.
- Pick three agents a , b and c from the population at random, they must be distinct from each other as well as from agent x
- Pick an index random index $R \in \{1, \dots, n\}$ where 'n' is the factor to be optimized. Compute the agent's potentially position new $y = [y_1, \dots, y_n]$ for each i , pick a uniformly distributed number

$$r_i = U(0,1) \text{ If } r_i < CR \text{ or } i = R \text{ then, set } y_i = a_i + F(b_i + c_i) \text{ otherwise set } y_i = x_i$$

- If $f(y) < f(x)$ then replace the agent in with the improved solution.

C. BIOGEOGRAPHY-BASED OPTIMIZATION

Biogeography-Based Optimization is a population based stochastic optimization technique based on the concept of nature's way of distribution of species [7]. Distribution of a species from one place to another is influenced by many factors such as rainfall, diversity of vegetation, relief forms, diversity of topographic features, land area; temperature etc. Movement of species from one area to another area results in sharing of their features with each other. Due to this movement, the quality of some species may improve due to exchange of good features with better species.

In context of biogeography, a habitat is defined as an island (area) that is geographically isolated from other islands. Geographical areas that are well suited as residences for biological species are said to the migration of some species from a habitat to an exterior habitat is known as emigration process and an entry into one habitat from an outside is known as immigration process.

The rate of immigration and the emigration are functions of the number of species in the habitat. Habitats with a high HSI have a low species immigration rate as they are already saturated with species [8]. This method is used where there is high dimension with multiple local optima and hence used to solve reactive power and OPF problem. Based on the immigration and migration values the total number of species in a habitat and their probability can be found.

D. TABU SEARCH ALGORITHM

Tabu search is a higher-level heuristic algorithm for solving combinatorial optimization problems. It is an iterative procedure starts from any initial solution and attempts to determine a better solution [9]. Generally, the advantages of this algorithm over other traditional optimization techniques can be summarized as follows:

- Tabu search is characterized by its ability to avoid entrapment in local optimal solution and prevent cycling by using flexible memory of search history.
- It can easily deal with non-smooth, non-continuous objective functions that are the real-life optimization problems.
- Tabu search uses probabilistic transition rule to make decisions, not deterministic rules.

This makes it more flexible and robust than conventional methods. The basic elements of this algorithm are briefly stated and defined as follows:

- **Current solution:** It is a set of the optimized parameter values at any iteration. It plays a central role in generating the neighbor trial solutions ($x_{current}$).
- **Moves:** They characterize the process of generating trial solutions that are related to the current solution ($x_{current}$).
- **Set of candidate moves:** $n(x_{current})$: It is the set of all possible moves or trial solutions, x_{trials} , in the neighborhood of $x_{current}$. In case of continuous variable optimization problems, this set is too large or even infinite set.
- **Tabu restrictions:** These are certain conditions imposed on moves that make some of them forbidden. These forbidden moves are listed to a certain size and known as Tabu. This list is called the Tabu list which is used to prevent cycling and avoid returning to the local optimum just visited.

- **Aspiration criterion (level):** It is a rule that overrides Tabu restrictions (if a certain move is forbidden by Tabu restriction, the aspiration criterion, when satisfied, can make this move allowable). The importance of using aspiration criterion is to add some flexibility in the algorithm by directing it towards the attractive moves [10].
- **Stopping criteria:** These are the conditions under which the search process will terminate. In this study, the search will terminate if one of the following criteria is satisfied, the number of iterations since the last change of the best solution is greater than a pre specified number (or) the number of iterations reaches the maximum allowable number.

E. MODIFIED SHUFFLED FROG LEAPING ALGORITHM

The RPO problem is a non linear optimization problem. The degree of nonlinearity causes difficulties in solving this problem using traditional methods. Therefore, one of the precise recently developed techniques called Modified Shuffled Frog Leaping Algorithm which is based on Shuffled Frog Leaping Algorithm has been explained in this paper. Shuffled Frog Leaping Algorithm is a decrease based stochastic search method that begins with an initial population of frogs whose characteristics, known as memes. The memes represent the decision variables [11].The algorithm consists of two main elements: local search and global information exchange

In this method, the total population is partitioned into groups. The groups are referred to as memplexes, which search independently. The population is divided into q memplexes which each containing p frogs. In each memplex, the frogs with the best and the worst fitness are identified as b_x and w_x , respectively. Also, the frog with the most qualified fitness level among

all the memplexes is identified as $g(x)$. Accordingly, the position of the frog with the worst fitness is adjusted this goal of the overall process is to determine global optimal solutions [12].

Step 1: The initial population for each iterate are randomly generated.

Step 2: The objective function value for each individual is calculated.

Step 3: The initial population based on the objective function values is sorted with decreasing manner.

Step 4: The population which is sorted is partitioned into memplexes by the following process, the first population goes to the first Memplex, the second population goes to the second memplex, population q th goes to the q th memplex, and population $q + 1$ goes back to the first memplex, etc [13].

Step 5: The best and worst population in each memplex is found and then generates the $b(x)$, $w(x)$ for them respectively.

Step 6: The frog with the global best fitness in all memplexes is identified as $g(x)$.

Step 7: A separate process is applied to improve only the frog with the worst fitness

Step 8: Once again all memplexes are combined and sorted again.

Step 9: Apply mutation in order to compensate if there is any drawback.

Step 10: If the current iteration number (iteration $max2$) reaches the predetermined maximum iteration number, the Search procedure is stopped, otherwise it goes to step 4.

Step 11: The last $g(x)$ is the solution of the problem.

F. PARTICLE SWARM OPTIMIZATION

The Particle swarm optimization algorithm is based on the ideas of social behavior of organisms such as animal flocking and fish Schooling. H.yoshida et al proposed a particle swarm optimization (PSO) for reactive power and voltage/var control (VVC) considering

voltage security assessment [14]. It determines an on-line VVC strategy with continuous and discrete control variables such as avr operating values of generators, tap positions of OLTC of transformers and the number of reactive power compensation equipment. Jong-bae park et al suggested a modified particle swarm optimization (MPSO) for economic dispatch with nonsmooth cost functions [15].

An improvisation has been made from the classical PSO and it is known as Modified particle swarm optimization (MPSO) algorithm. In the new algorithm, particles not only studies from itself and the best one but also from other individuals. By this enhanced study behavior, the opportunity to find the global optimum is increased and the influence of the initial position of the particles is decreased [16]. PSO is initialized with a group of random particles and then it searches for optima by updating generations. In every iteration each particle is updated by following “two best” values. The first one is the best solution (fitness value) it has achieved so far. This value is called $pbest$. Another best value that is tracked by the particle swarm optimizer is the global best called $gbest$. After finding the best values the particles update its velocity and position. Implementation of an optimization problem of Genetic Algorithm is realized within the evolutionary process of a fitness function. The fitness function adopted is given as:

$$\text{Fitness function} = \frac{1}{(\text{objective} + \text{penalty})}$$

Where objective function is the generation cost and the penalty is the bus voltage angle. Penalty cost has been added to discourage solutions which violate the binding constraints. Finally, the penalty factor is tended to zero.

The PSO algorithm to solve the optimal power flow can be summarized as follows:

Step 1: The population is initialized according to the constraints.

Step 2: For each individual in the population, the fitness function is evaluated in de -normalized form.

Step 3: The velocity is updated and new population is created

Step 4: If maximum iteration number is reached, then go to next step else go step2.

Step 5: Print the best individual's settings.

G. HARMONY SEARCH ALGORITHM

The harmony search algorithm was proposed by Geem [17]. It is a nature inspired algorithm, which mimics the improvisation of music players. The harmony in music is analogous to the optimization solution vector, and the musician's improvisations are analogous to the local and global search schemes in optimization techniques. The HS algorithm uses a stochastic random search, instead of a gradient search .It uses only simple parameters and it is very easy for implementation .The algorithm includes the following steps.

- 1) The optimization problem parameters are initialized.
- 2) Initialize the harmony memory.
- 3) A New Harmony memory is formed by improvisation.
- 4) The harmony memory is updated.
- 5) Check for stopping criteria. Otherwise, repeat step 3 to 4.
- 6) The non dominated solution vectors in this algorithm are known as Pareto optimal solutions.
- 7) Best compromise solution vector is taken from the Pareto optimal set using fuzzy approach [18].

H. ARTIFICIAL BEE COLONY ALGORITHM

Swarm intelligence is a form of artificial intelligence (AI) based on the collective behaviors of animals or certain phenomenon of natural systems such as ants, fish and birds. Recently, a novel swarm-based intelligence

called the bees algorithm, which mimics the food foraging behavior of honey bees colony, has been developed and it is considered to be as efficient as other swarm intelligence approaches [19]. This approach is used to solve RPO problem .The process of food foraging starts from sending out a group of scout bees to search for flower patches that contain a large amount of nectar and pollen. After returning to their hive, those scout bees would perform a special movement, which is known as the "waggle dance" to communicate with other bees and report three kinds of information regarding the flower patches, which are the direction of food sources, their quality and distances. This information helps the bees travelling towards flower patches more rapidly and precisely without using guides or maps. More follower bees are sent to more promising patches and allow the colony to collect food during the short amount of time. Additionally, the bees evaluate the food level from the flower patch to decide how to perform the next waggle dance and recruit more workers to the remaining food sources. The food foraging process will repeat as long as there is enough food sources in the search space, and more bees will be recruited to collect food more effectively [20]. To solve RPO problem this algorithm involves the following steps:

- The bee algorithm parameters are initialized with random solutions.
- Evaluate the fitness of the population and select the number of best sites for neighborhood search.
- Select the best number of bees in best sites and calculate its fitness value.
- Assign remaining bees to search randomly in the search space and evaluate the fitness value.
- If the stopping criterion is met the optimal value can be found else assign a new

population of scout bees and repeat the steps.

Additionally, the remaining bees are assigned to search randomly in the search space in order to find optimal solutions

I. ANT COLONY OPTIMIZATION

Ant colony search is one of the recently developed optimization algorithm to solve RPO problem. This algorithm is developed based on the behavior of real ant colonies that are used to solve function or combinatorial optimization problems. This system was introduced by Marco Dorigo and was called “ant system” [21]. As it is well known, real ants are capable of finding the shortest path from food sources. They are also capable of adapting to changes in the surrounding environment.

The studies regarding the behavior of ants reveal that such capabilities are essentially due to what is called “pheromone trails”, which ants use to communicate information among individuals regarding path and to decide where to go [22]. The pheromone trails directs the ants towards food path. The process can be explained as follows:

- (a) Real ants follow a path between nest and food source.
- (b) An obstacle appears on the path: ants choose whether to turn left or right with equal probability.
- (c) Pheromone is deposited more quickly on the shorter path.
- (d) All ants have chosen the shorter path.

Thus all the ants follow the shortest path and reach the food source, similarly this algorithm can be applied for optimization problem by following the same technique.

J. GRAVITATIONAL SEARCH ALGORITHM

Gravitational Search Algorithm (GSA) is the new meta-heuristic optimization algorithm motivated by the Newton’s laws of gravity and motion. GSA was firstly produced by Rashedi et

al. in 2009. According to this algorithm, agents are considered as objects and their performance is measured by their masses. Every object attracts every other object with gravitational force. This algorithm is highly efficient to solve RPO problem [23]. It involves the following steps

- Initialization of parameters and the position of masses are fixed randomly.
- The best and worst fitness is evaluated for all agents and the gravitational constant G is calculated as $G(t) = G_0 \exp(-\alpha \frac{t}{T})$, Where G_0 is the initial value of the Gravitational constant, α is a constant t is the current epoch and T is the total iteration number.
- Update the Gravitational and Inertial Masses and calculate the total force acting on the agents.
- Update the position of the agents using the formula $x_i^d(t+1) = x_i^d t + v_i^d(t+1)$.
- Repeat the iterations until the stopping criteria are met.

Thus the parameters considered for reactive power optimization can be solved effectively by this algorithm

CONCLUSION

In this paper an attempt has been made to review various optimization methods used to solve RPO problems. Even though, excellent advancements have been made in classical methods, they suffer with the following disadvantages: In most cases, mathematical formulations have to be simplified to get the solutions because of the extremely limited capability to solve real-world large-scale power system problems. They are weak in handling qualitative constraints. They have poor convergence, may get stuck at local optimum, they can find only a single optimized solution in a single simulation run, they become too slow if number of variables are large and they are

computationally expensive for solution of a large system constraints.

ACKNOWLEDGEMENT

Authors acknowledge the immense help received from the scholars whose articles are cited and included in references of this manuscript. Authors acknowledge the kind support from the institution Sri Krishna College Of Technology, Coimbatore.

REFERENCES

1. J.J. Grainger and W.D Stevenson, "Power System Analysis", McGraw-Hill, New York, 1994, ISBN 0-07-061293-5.
2. Zhanhong Wei Zhihua Cui and Jianchao Zeng, "Social Cognitive Optimization Algorithm with Reactive Power Optimization of Power System," 2010 *International Conference on Computational Aspects of Social Networks*, 978-0-7695-4202-7/10 \$26.00 © 2010 IEEE DOI 10.1109/CASoN.2010.10.
3. K. Lenin, M. R. Mohan, "Ant Colony Search Algorithm for Optimal Reactive Power Optimization", *Serbian Journal of Electrical Engineering*, Vol. 3, No. 1, June 2006, 77 - 88.
4. William W. Hager and Hongchao Zhang, "A New Conjugate Gradient Method with Guaranteed Descent and an Efficient Line Search", *SIAM Journal on Optimization*. C_ 2005 *Society for Industrial and Applied Mathematics*, Vol. 16, No. 1, pp. 170–192.
5. R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces", Technical Report Tr-95-012, March 1995.
6. Brest.J, Greiner.S, Boskovic.B, Mernik.M, Zumer.V, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark functions", *IEEE Transactions on Evolutionary Computation*, VOL. 10, NO. 6, DECEMBER 2006.
7. Dan Simon, "Biogeography-Based Optimization", *IEEE Transaction On evolutionary Computation*, Vol. 12, No. 6, pp. 702-713, December 2008.
8. Rick Rarick, Dan Simon, F. Eugenio Villaseca, Bharat Vyakaranam.J, "Biogeography-Based Optimization and the Solution of the Power Flow Problem", *Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics*, San Antonio, TX, USA - October 2009.
9. M. A. Abido, "Optimal Power Flow Using Tabu Search Algorithm", *Electric Power Components and Systems*, Taylor & Francis, 2002.
10. Whei-Min Lin, Fu-Sheng Cheng, and Ming-Tong Tsay, "An Improved Tabu Search for Economic Dispatch with Multiple Minima," *IEEE Transactions on Power Systems*, Vol. 17, No. 1, February 2002.
11. H.A. Shayanfar, R. Jahani, J. Olamaei, "Comparison of Modified Shuffled Frog Leaping Algorithm and Other Heuristic Methods for Optimal Placement of Unified Power Flow Controllers in Electrical Power System", *Australian Journal of Basic and Applied Sciences*, Vol 4, No:11, 2010.
12. Mohammad Rasoul Narimani, "A New Modified Shuffle Frog Leaping Algorithm for Non-Smooth Economic Dispatch", *World Applied Sciences Journal*, Vol .12, No: 6, 2011.
13. Muzaffar M. Eusuff and Kevin E. Lansey, "Optimization of Water Distribution Network Design Using the Shuffled Frog Leaping Algorithm", *Journal Of Water Resources Planning And Management*, \Vol. 129, No. 3, May 1, 2003.
14. H. Yoshida, K. Kawata, Y. Fukuyama Et Al, "A Particle Swarm Optimization for

- Reactive Power and Voltage Control Considering Voltage Security Assessment”, IEEE Trans. Power 2000.
15. J. B. Park, Ki. S. Lee, J. R. Shi And K. Y. Lee, “A Particle Swarm Optimization For Economic Dispatch With Nonsmooth Cost Functions”, IEEE Trans. Power Syst., Vol. 20, No. 1, Pp. 34-42, Feb. 2005.
 16. Cui-Ru Wang, He-Jin Yuan, Zhi-Qiang Huang, Jiang-Wei Zhang, Chen-Jun Sun, “A Modified Particle Swarm Optimization Algorithm And Its Application In Optimal Power Flow Problem”, *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, 18-21 August 2005.
 17. Zong Woo Geem, “Harmony Search Algorithm for Solving Sudoku”, Part I, LNAI 4692, pp. 371–378, 2007, ©Springer-Verlag Berlin Heidelberg 2007.
 18. K. Guney, M. Onay, “Optimal synthesis of linear antenna arrays using a harmony search algorithm”, *Expert Systems and Applications*, 2011, Elsevier Ltd.
 19. Dervis Karaboga and Bahriye Basturk, “Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems”, IFSA 2007, LNAI 4529, pp. 789–798, Springer-Verlag Berlin Heidelberg, 2007.
 20. D. Karaboga, B. Akay, “Artificial Bee Colony (ABC), Harmony Search and Bees Algorithms on Numerical Optimization”, *Innovative Productions Machines and Systems, Proceedings of IPROMS 2009 Conference*, 2009.
 21. Marco Dorigo, Gianni Di Caro, “Ant Colony Optimization: A New Meta-Heuristic”, *Proceedings of the IEEE Transactions*, 1999.
 22. Johann Dr’eo and Patrick Siarry, “A New Ant Colony Algorithm Using the Heterarchical Concept Aimed at Optimization of Multim minima Continuous Functions”, *Springer link journal of Computer Science*, 2002, Volume 2463/2002, 216-221.
 23. Serhat Duman, Yusuf Sonmez, Ugur Guvenc, “Application of Gravitational Search Algorithm for Optimal Reactive Power Dispatch Problem”, *Transactions of IEEE journal*, 2011.