



Science

ACTIVE POWER LOSS REDUCTION BY AMPLIFIED ANT COLONY ALGORITHM

Dr.K. Lenin ^{*1}

^{*1} Professor, Department of EEE, Prasad V. Potluri Siddhartha Institute of Technology, Kanuru, Vijayawada, Andhra Pradesh -520007, India

Abstract

In this paper, Amplified Ant Colony (AAC) algorithm has been proposed for solving optimal reactive power problem. Mutation of Genetic algorithm (GA) is used in Ant Colony Algorithm (ACA) and the output of the GA is given as an input to the ACA. The proposed Amplified Ant Colony (AAC) algorithm has been tested on standard IEEE 118 & practical 191 bus test systems and simulation results show clearly the superior performance of the proposed Amplified Ant Colony (AAC) algorithm in reducing the real power loss & voltage profiles are within the limits.

Keywords: Optimal Reactive Power; Transmission Loss; Genetic Algorithm; Ant Colony Algorithm.

Cite This Article: Dr.K. Lenin. (2018). “ACTIVE POWER LOSS REDUCTION BY AMPLIFIED ANT COLONY ALGORITHM.” *International Journal of Research - Granthaalayah*, 6(7), 132-141. <https://doi.org/10.5281/zenodo.1323035>.

1. Introduction

The main objective of optimal reactive power problem is to minimize the real power loss and bus voltage deviation. Various numerical methods like the gradient method [1-2], Newton method [3] and linear programming [4-7] have been adopted to solve the optimal reactive power dispatch problem. Both the gradient and Newton methods have the complexity in managing inequality constraints. If linear programming is applied then the input- output function has to be uttered as a set of linear functions which mostly lead to loss of accuracy. The problem of voltage stability and collapse play a major role in power system planning and operation [8]. Evolutionary algorithms such as genetic algorithm have been already proposed to solve the reactive power flow problem [9-11]. Evolutionary algorithm is a heuristic approach used for minimization problems by utilizing nonlinear and non-differentiable continuous space functions. In [12], Hybrid differential evolution algorithm is proposed to improve the voltage stability index. In [13] Biogeography Based algorithm is projected to solve the reactive power dispatch problem. In [14], a fuzzy based method is used to solve the optimal reactive power scheduling method. In [15], an improved evolutionary programming is used to solve the optimal reactive power dispatch problem. In [16], the optimal reactive power flow problem is solved by integrating a genetic algorithm with a nonlinear interior

point method. In [17], a pattern algorithm is used to solve ac-dc optimal reactive power flow model with the generator capability limits. In [18], F. Capitanescu proposes a two-step approach to evaluate Reactive power reserves with respect to operating constraints and voltage stability. In [19], a programming based approach is used to solve the optimal reactive power dispatch problem. In [20], A. Kargarian et al present a probabilistic algorithm for optimal reactive power provision in hybrid electricity markets with uncertain loads. This paper introduces a novel Enhanced Ant colony Algorithm (EACA) for solving optimal reactive power dispatchpower problem. Genetic Algorithm (GA) and Ant Colony Algorithm (ACA) [21-26] has been combined to solve the reactive power problem. The proposed Amplified Ant Colony (AAC) algorithm has been tested on standard IEEE 118 & practical 191 bus test systems and simulation results show clearly the superior performance of the proposed Amplified Ant Colony (AAC) algorithm in reducing the real power loss & voltage profiles are within the limits.

2. Problem Formulation

2.1. Active Power Loss

The objective of the reactive power dispatch is to minimize the active power loss in the transmission network, which can be described as follows:

$$F = PL = \sum_{k \in Nbr} g_k (V_i^2 + V_j^2 - 2V_i V_j \cos \theta_{ij}) \quad (1)$$

Or

$$F = PL = \sum_{i \in Ng} P_{gi} - P_d = P_{gslack} + \sum_{i \neq slack}^{Ng} P_{gi} - P_d \quad (2)$$

Where g_k : is the conductance of branch between nodes i and j , Nbr : is the total number of transmission lines in power systems. P_d : is the total active power demand, P_{gi} : is the generator active power of unit i , and P_{gslack} : is the generator active power of slack bus.

2.2. Voltage Profile Improvement

For minimizing the voltage deviation in PQ buses, the objective function becomes:

$$F = PL + \omega_v \times VD \quad (3)$$

Where ω_v : is a weighting factor of voltage deviation.

VD is the voltage deviation given by:

$$VD = \sum_{i=1}^{Npq} |V_i - 1| \quad (4)$$

2.3. Equality Constraint

The equality constraint of the optimal reactive power dispatch power (ORPD) problem is represented by the power balance equation, where the total power generation must cover the total power demand and the power losses:

$$P_G = P_D + P_L \quad (5)$$

This equation is solved by running Newton Raphson load flow method, by calculating the active power of slack bus to determine active power loss.

2.4. Inequality Constraints

The inequality constraints reflect the limits on components in the power system as well as the limits created to ensure system security. Upper and lower bounds on the active power of slack bus, and reactive power of generators:

$$P_{gslack}^{min} \leq P_{gslack} \leq P_{gslack}^{max} \quad (6)$$

$$Q_{gi}^{min} \leq Q_{gi} \leq Q_{gi}^{max}, i \in N_g \quad (7)$$

Upper and lower bounds on the bus voltage magnitudes:

$$V_i^{min} \leq V_i \leq V_i^{max}, i \in N \quad (8)$$

Upper and lower bounds on the transformers tap ratios:

$$T_i^{min} \leq T_i \leq T_i^{max}, i \in N_T \quad (9)$$

Upper and lower bounds on the compensators reactive powers:

$$Q_c^{min} \leq Q_c \leq Q_c^{max}, i \in N_C \quad (10)$$

Where N is the total number of buses, N_T is the total number of Transformers; N_C is the total number of shunt reactive compensators.

3. Genetic Algorithms

Genetic algorithms (GA's) are search algorithms that work via the process of natural selection. They begin with a sample set of potential solutions which then evolves toward a set of more optimal solutions. Within the sample set, solutions that are poor tend to die out while better solutions mate and propagate their advantageous traits, thus introducing more solutions into the set that boast greater potential (the total set size remains constant; for each new solution added, an old one is removed). A little random mutation helps guarantee that a set won't stagnate and simply fill up with numerous copies of the same solution.

In general, genetic algorithms tend to work better than traditional optimization algorithms because they're less likely to be led astray by local optima. This is because they don't make use of single-point transition rules to move from one single instance in the solution space to another. Instead, GA's take advantage of an entire set of solutions spread throughout the solution space, all of which are experimenting upon many potential optima.

However, in order for genetic algorithms to work effectively, a few criteria must be met:

- It must be relatively easy to evaluate how "good" a potential solution is relative to other potential solutions.
- It must be possible to break a potential solution into discrete parts that can vary independently. These parts become the "genes" in the genetic algorithm.
- Finally, genetic algorithms are best suited for situations where a "good" answer will suffice, even if it's not the absolute best answer.

Basic Mechanics of Genetic Algorithms

The basic operations of the genetic algorithm are simple and straight-forward:

Reproduction: The act of making a copy of a potential solution:

Crossover: The act of swapping gene values between two potential solutions, simulating the "mating" of the two solutions.

Mutation: The act of randomly altering the value of a gene in a potential solution.

Fitness Functions and Natural Selection

As mentioned earlier, it's necessary to be able to evaluate how "good" a potential solution is relative to other potential solutions. The "fitness function" is responsible for performing this evaluation and returning a positive integer number, or "fitness value", that reflects how optimal the solution is: the higher the number, the better the solution.

The fitness values are then used in a process of natural selection to choose which potential solutions will continue on to the next generation, and which will die out. It should be noted, however, that natural selection process does not merely choose the top x number of solutions; the solutions are instead chosen statistically such that it is more likely that a solution with a higher fitness value will be chosen, but it is not guaranteed. This tends to correspond to the natural world.

A common metaphor for the selection process is that of a large roulette wheel. Remembering that fitness values are positive integers, imagine that each potential solution gets a number of slots on the wheel equal to its fitness value. Then the wheel is spun and the solution on which it stops is selected. Statistically speaking, solutions with a higher fitness value will have a greater chance of being selected since they occupy more slots on the wheel, but even solutions with just a single slot still have a chance.

4. Ant Colony Algorithm

Based on the fact that ants are able to find the shortest route between their nest and a source of food. This is done using pheromone trails, which ants deposit whenever they travel, as a form of indirect communication. when ants leave their nest to search for a food source, they randomly rotate around an obstacle, and initially the pheromone deposits will be the same for the right and

left directions. When the ants in the shorter direction find a food source, they carry the food and start returning back, following their pheromone trails, and still depositing more pheromone. An ant will most likely choose the shortest path when returning back to the nest with food as this path will have the most deposited pheromone. For the same reason, new ants that later starts out from the nest to find food will also choose the shortest path. Over time, this positive feedback process prompts all ants to choose the shorter path.

Ant chooses direction using roulette wheel selection applied to the pheromone amounts. Than it moves arbitrarily to the place the vector points and does an arbitrary walk inside a local search radius. This radius can shrink in time to do more detailed searches around the point. The ant Fitness is then calculated from its position and the amount of pheromone on choosing direction vector is proportionally increased. If a better solution is found, the vector is changed to actual ant position.

5. Amplified Ant Colony (AAC) algorithm

In this phase, the ant colony algorithm is used as a mutation of GA, the output of the GA is given as an input to the ACA. The genetic algorithm undergoes the selection, crossover process and it gives the result. The result contains only one value which is optimal value. This is the procedure utilized in, Amplified Ant Colony (AAC) algorithm.

- Preliminary population

Genetic algorithm initiate its work with a group of chromosomes (solutions) known as the preliminary population, members of this population be configured up arbitrarily.

- Fitness Function

This step is the appraisal of solutions (chromosome).

- Creation of new generations

Selection

There are many different selection methods, such as elitist selection, rank selection and roulette wheel selection. In this paper, the roulette wheel selection technique is used. In roulette wheel selection, the individual is selected based on the relative fitness with its contestants. This is alike to dividing the wheel into a number of slices.

Crossover

There are many different crossover methods, in this paper, the arithmetic crossover is used. Arithmetic crossover operator linearly combines two parent chromosome vectors to create two new offspring according to the equations:

$$\text{offspring 1} = a * \text{parent1} + (1 - a) * \text{parent2} \quad (11)$$

$$\text{offspring 2} = (1 - a) * \text{parent1} + a * \text{parent2} \quad (12)$$

Where is “a” arbitrary weighting factor chosen before each crossover operation.

Using ACA as a mutation for GA according to the following steps

- 1) Chromosomes that have been obtained from the procedure of mating (crossover) is pass in into ant colony algorithm, is calculate the probability (P_{ij}) of each chromosome according to the equation Probability,

$$P_{ij} = \frac{T_{ij}}{\sum_{j=1}^P T_{ij}} \quad (13)$$

Where T_{ij} is an amounts of pheromone, $T_{ij}=1$ for each chromosome (solution) in preliminary state of algorithm, P number of chromosome, Set the iteration number $L=1$.

- 2) Create arbitrary numbers in the range (0,1), one for each ant. Number of ants= number of chromosomes.
- 3) Each ant chooses specific value (specific chromosome) if the ant be within the range of probability of the value (chromosome), this step is repeated for each ant.
- 4) Calculate the objective function values; determine the best and worst values among the values chosen by different ants.

$F_{best} = \min(\text{values})$

$F_{worst} = \max(\text{values})$

The greatest value is stored during each iteration within a matrix.

- 5) Test for the convergence. The procedure is assumed to converge if all ants take the same best value. If convergence is not attained, then assume that all the ants return home and start again in search of food. Set the new iteration number as $L = L + 1$, and modernize the pheromones on different values as,

$$T_{ij}^{(L)} = T_{ij}^{(old)} + \sum_K \Delta T^{(K)} \quad (14)$$

Where $T_{ij}^{(old)}$ symbolizes the pheromone amount of the preceding iteration left after evaporation, which is taken as,

$$T_{ij}^{(old)} = (1 - PP) * T_{ij}^{(L-1)} \quad (15)$$

and $\sum_K \Delta T^{(K)}$ is the pheromone deposited by the best ant k and the summation extends over all the best ants k (if multiple ants take the same best path). The evaporation rate or pheromone decay factor PP is assumed to be in the range 0.48 to 0.79 and the pheromone deposited $\sum_K \Delta T^{(K)}$ is computed using,

$$\sum_K \Delta T^{(K)} = \frac{F_{best}}{F_{worst}} \quad (16)$$

With the new values of $T_{ij}^{(L)}$, go to step 1. Steps

1, 2, 3, 4 and 5 are repeated until the process converges, that is, until all the ants select the same best path. In some cases, the iterative process is stopped after completing a pre-specified maximum number of iterations (L max).

- 6) Modernized generation: The matrix contains that best value is enter to genetic algorithm as population.
- 7) End Criterion: The iterative process is stopped after completing a pre-specified maximum number of generations of genetic algorithm.

6. Simulation Results

At first Amplified Ant Colony (AAC) algorithm has been tested in standard IEEE 118-bus test system [27]. The system has 54 generator buses, 64 load buses, 186 branches and 9 of them are with the tap setting transformers. The limits of voltage on generator buses are 0.95 -1.1 per-unit., and on load buses are 0.95 -1.05 per-unit. The limit of transformer rate is 0.9 -1.1, with the changes step of 0.025. The limitations of reactive power source are listed in Table 1, with the change in step of 0.01.

Table 1: Limitation of reactive power sources

BUS	5	34	37	44	45	46	48
QCMAX	0	14	0	10	10	10	15
QCMIN	-40	0	-25	0	0	0	0
BUS	74	79	82	83	105	107	110
QCMAX	12	20	20	10	20	6	6
QCMIN	0	0	0	0	0	0	0

The statistical comparison results have been listed in Table 2 and the results clearly show the better performance of proposed Amplified Ant Colony (AAC) algorithm.

Table 2: Comparison results

Active power loss (p.u)	BBO [28]	ILSBBO/strategy1 [28]	ILSBBO/strategy1 [28]	Proposed AAC
Min	128.77	126.98	124.78	115.68
Max	132.64	137.34	132.39	120.12
Average	130.21	130.37	129.22	118.24

Then the Amplified Ant Colony (AAC) algorithm has been tested in practical 191 test system and the following results have been obtained. In Practical 191 test bus system – Number of Generators = 20, Number of lines = 200, Number of buses = 191 Number of transmission lines = 55. Table 3 shows the optimal control values of practical 191 test system obtained by AAC method. And table 4 shows the results about the value of the real power loss by obtained by Amplified Ant Colony (AAC) algorithm.

Table 3: Optimal Control values of Practical 191 utility (Indian) system by AAC method

VG1	1.100	VG 11	0.900
VG 2	0.760	VG 12	1.000
VG 3	1.010	VG 13	1.000
VG 4	1.010	VG 14	0.900
VG 5	1.100	VG 15	1.000

VG 6	1.100	VG 16	1.000
VG 7	1.100	VG 17	0.900
VG 8	1.010	VG 18	1.000
VG 9	1.100	VG 19	1.100
VG 10	1.010	VG 20	1.100

T1	1.000	T21	0.900	T41	0.900
T2	1.000	T22	0.900	T42	0.900
T3	1.000	T23	0.900	T43	0.910
T4	1.100	T24	0.900	T44	0.910
T5	1.000	T25	0.900	T45	0.910
T6	1.000	T26	1.000	T46	0.900
T7	1.000	T27	0.900	T47	0.910
T8	1.010	T28	0.900	T48	1.000
T9	1.000	T29	1.010	T49	0.900
T10	1.000	T30	0.900	T50	0.900
T11	0.900	T31	0.900	T51	0.900
T12	1.000	T32	0.900	T52	0.900
T13	1.010	T33	1.010	T53	1.000
T14	1.010	T34	0.900	T54	0.900
T15	1.010	T35	0.900	T55	0.900
T19	1.020	T39	0.900		
T20	1.010	T40	0.900		

Table 4: Optimum real power loss values obtained for practical 191 utility (Indian) system by AAC method

Real power Loss (MW)	AAC
Min	140.226
Max	145.048
Average	143.124

7. Conclusion

In this paper, Amplified Ant Colony (AAC) algorithm has been efficiently implemented to solve optimal Reactive Power problem. *Mutation of Genetic algorithm (GA) is used in Ant Colony Algorithm (ACA) and the output of the GA is given as an input to the ACA.* The proposed Amplified Ant Colony (AAC) algorithm has been tested on standard IEEE 118 & practical 191 bus test systems and simulation results show clearly the superior performance of the proposed Amplified Ant Colony (AAC) algorithm in reducing the real power loss & voltage profiles are within the limits.

References

- [1] O.Alsac, B. Scott, "Optimal load flow with steady state security", IEEE Transaction. PAS -1973, pp. 745-751.
- [2] Lee K Y, Paru Y M, Ortiz J L –A united approach to optimal real and reactive power dispatch, IEEE Transactions on power Apparatus and systems 1985: PAS-104 : 1147-1153

- [3] 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.
- [4] Deeb N, Shahidehpur S.M, Linear reactive power optimization in a large power network using the decomposition approach. IEEE Transactions on power system 1990: 5(2): 428-435
- [5] E. Hobson, 'Network constrained reactive power control using linear programming, 'IEEE Transactions on power systems PAS -99 (4), pp 868=877, 1980
- [6] K.Y Lee, Y.M Park, and J.L Ortiz, “Fuel –cost optimization for both real and reactive power dispatches”, IEE Proc; 131C, (3), pp.85-93.
- [7] M.K. Mangoli, and K.Y. Lee, “Optimal real and reactive power control using linear programming”, Electr.Power Syst.Res, Vol.26, pp.1-10,1993.
- [8] C.A. Canizares, A.C.Z.de Souza and V.H. Quintana, “Comparison of performance indices for detection of proximity to voltage collapse,” vol. 11. no.3, pp.1441-1450, Aug 1996.
- [9] K.Anburaja, “Optimal power flow using refined genetic algorithm”, Electr.Power Compon.Syst , Vol. 30, 1055-1063,2002.
- [10] D. Devaraj, and B. Yeganarayana, “Genetic algorithm based optimal power flow for security enhancement”, IEE proc-Generation.Transmission and. Distribution; 152, 6 November 2005.
- [11] A. Berizzi, C. Bovo, M. Merlo, and M. Delfanti, “A ga approach tocompare orpf objective functions including secondary voltage regulation,”Electric Power Systems Research, vol. 84, no. 1, pp. 187 – 194, 2012.
- [12] C.-F. Yang, G. G. Lai, C.-H. Lee, C.-T. Su, and G. W. Chang, “Optimalsetting of reactive compensation devices with an improved voltage stability index for voltage stability enhancement,” International Journalof Electrical Power and Energy Systems, vol. 37, no. 1, pp. 50 – 57,2012.
- [13] P. Roy, S. Ghoshal, and S. Thakur, “Optimal var control for improvementsin voltage profiles and for real power loss minimization usingbiogeography based optimization,” International Journal of ElectricalPower and Energy Systems, vol. 43, no. 1, pp. 830 – 838, 2012.
- [14] B. Venkatesh, G. Sadasivam, and M. Khan, “A new optimal reactivepower scheduling method for loss minimization and voltage stabilitymargin maximization using successive multi-objective fuzzy lp technique,”IEEE Transactions on Power Systems, vol. 15, no. 2, pp. 844 –851, may 2000.
- [15] W. Yan, S. Lu, and D. Yu, “A novel optimal reactive power dispatchmethod based on an improved hybrid evolutionary programming technique,”IEEE Transactions on Power Systems, vol. 19, no. 2, pp. 913 –918, may 2004.
- [16] W. Yan, F. Liu, C. Chung, and K. Wong, “A hybrid genetic algorithm interior point method for optimal reactive power flow,” IEEE Transactions on Power Systems, vol. 21, no. 3, pp. 1163 – 1169, aug. 2006.
- [17] J. Yu, W. Yan, W. Li, C. Chung, and K. Wong, “An unfixed piecewise optimal reactive power-flow model and its algorithm for ac-dc systems,”IEEE Transactions on Power Systems, vol. 23, no. 1, pp. 170 –176, feb.2008.
- [18] F. Capitanescu, “Assessing reactive power reserves with respect tooperating constraints and voltage stability,” IEEE Transactions on Power Systems, vol. 26, no. 4, pp. 2224–2234, nov. 2011.
- [19] Z. Hu, X. Wang, and G. Taylor, “Stochastic optimal reactive power dispatch: Formulation and solution method,” International Journal of Electrical Power and Energy Systems, vol. 32, no. 6, pp. 615 – 621,2010.
- [20] A. Kargarian, M. Raoofat, and M. Mohammadi, “Probabilistic reactivepower procurement in hybrid electricity markets with uncertain loads,”Electric Power Systems Research, vol. 82, no. 1, pp. 68 – 80, 2012.
- [21] Elbeltagi, E., Hegazy, T., Grierson, D., (2005), "Comparison among five evolutionary-basedoptimization algorithms ", AdvancedEngineering Informatics.
- [22] Gao, S., Zhang, Z., and Cao. C., (2010), " ANovel Ant Colony Genetic Hybrid Algorithm",Journal of Software, Vol. 5.

- [23] Aravindh, S. and Michael, G., (2012)," Hybrid of Ant Colony Optimization and Genetic Algorithm for Shortest Path in Wireless Mesh Networks ", Journal of Global Research in Computer Science, Vol. 3.
- [24] Sivanaudom, S.N., Deepa, S.N. (2008), "An Introduction To Genetic Algorithms ", Springer–Verlag Berlin Heidelberg, India.
- [25] Aidov, A., Dulikravich, G. S., (2009),"Modified Continuous Ant Colony Algorithm ", 2nd International Congress of Serbian Society of Mechanics.
- [26] Kovarik, O., (2006), "Ant Colony Optimization for Continuous Problems", Msc. Thesis, Dept. of Electrical Engineering, University of Czech Technical.
- [27] IEEE, "The IEEE 30-bus test system and the IEEE 118-test system", (1993), <http://www.ee.washington.edu/trsearch/pstca/>.
- [28] Jiangtao Cao, Fuli Wang and Ping Li, "An Improved Biogeography-based Optimization Algorithm for Optimal Reactive Power Flow", International Journal of Control and Automation Vol.7, No.3 (2014), pp.161-176.

*Corresponding author.

E-mail address: gklenin@ gmail.com