Self Adaptive Firefly Algorithm for Economic Load Dispatch

Dr. B. Suresh Babu
Professor, Electrical and Electronics Engineering, Vaagdevi College of Engineering
Bollikunta, Warangal – 506005, Telangana State

Abstract – *Economic load dispatch (ELD) is an important operational problem of the power system, aiming to minimize the fuel cost. The firefly algorithm (FA), a heuristic numeric optimization algorithm inspired by the behavior of fireflies, appears to be a robust and reliable technique. This paper presents a self adaptive FA for the solution of the ELD problem. The proposed algorithm (PA) is applied to the standard IEEE 14 and 30 bus test systems and the results are presented to demonstrate its effectiveness.*

Keywords: *economic load dispatch, firefly algorithm.*

NOMENCLATURE

a_i b_i c_i	fuel cost coefficients
d_i e_i	coefficients of valve point effects of the generator
ELD	economic load dispatch
FA	firefly algorithm
F_C	net fuel cost
I_i	light intensity of the i -th firefly
$Iter^{max}$	maximum number of iterations for convergence check.
nd	number of decision variables
nf	number of fireflies in the populations
ng	number of generators
PA	proposed algorithm
$p(V,\delta)$	real bus power as a function of voltage magnitude and voltage angle
$q(V,\delta)$	reactive bus power as a function of voltage magnitude and voltage angle
P_{Gi} and Q_{Gi}	real and reactive power generation at i -th bus respectively
P_{Di} and Q_{Di}	real and reactive power demand at i -th bus respectively
P_{Gi}^{min} and P_{Gi}^{max}	lower and upper limits of P_{Gi}
Q_{Gi}^{min} and Q_{Gi}^{max}	lower and upper limits of Q_{Gi}
r_{ij}	Cartesian distance between the i -th and j -th firefly
SAFA	self adaptive FA
t	iteration counter
x_i	i -th firefly
$\beta_{i,j}$	attractiveness between the i -th and j -th firefly
β_o and γ	maximum attractiveness and light intensity absorption coefficient respectively

I. INTRODUCTION

Present day power systems have the problem of deciding how best to meet the varying power demand that has a daily and weekly cycle in order to maintain a high degree of economy and reliability. Among the options that are available for an engineer in choosing how to operate the system, economic load dispatch (ELD) is the most significant. ELD is a computational process whereby the total required generation is distributed among the generating units in operation so as to minimize the total generation cost, subject to load and operational constraints. The objective of ELD is to minimize the total generation cost of a power system for a given load while satisfying various constraints [1].

Over the years numerous methods with various degrees of near-optimality, efficiency, ability to handle difficult constraints and heuristics, are suggested in the literature for solving the dispatch problems. These problems are traditionally solved using mathematical programming techniques such as lambda iteration method, gradient method, linear programming, dynamic programming method and so on. Many of these methods suffer from natural complexity and converge slowly. However, the classical lambda-iteration method has been in use for a long time. The additional constraints such as line flow limits cannot be included in the lambda iteration approach and the convergence of the iterations is dependent on the initial choice of lambda. In large power systems, this method has oscillatory problems that increase the computation time [1,2].

Apart from the above methods, there is another class of numerical techniques called evolutionary search algorithms such as simulated annealing, genetic algorithms, evolutionary programming, ant colony, artificial bee colony and particle swarm optimization have been applied in solving ELD [3-8]. Having in common processes of natural evolution, these algorithms share many similarities; each maintains a population of solutions that are evolved through random alterations and selection. The differences between these procedures lie in the representation techniques they utilize to encode candidates, the type of alterations they use to create new solutions, and the mechanism they employ for selecting the new parents. The algorithms have yielded satisfactory results across a great variety of power system problems. The main difficulty is their sensitivity to the choice of the parameters, such as temperature in SA, the crossover and mutation probabilities in GA and the inertia weight, acceleration coefficients and velocity limits in PSO.

Recently, firefly algorithm (FA) has been suggested for solving optimization problems [9,10]. It is inspired by the light attenuation over the distance and fireflies' mutual attraction rather than the phenomenon of the fireflies' light flashing. In this approach, each problem solution is represented by a firefly, which tries to move to a greater light source, than its own. It has been applied to a variety of ELD problems [11-14] and found to yield satisfactory results. However, the choice of FA parameters is important in obtaining good convergence and global optimal solution.

A self adaptive FA (SAFA) for obtaining the global best solution has been suggested in this paper. The proposed algorithm (PA) has been tested on the IEEE 14 and 30 bus test systems to illustrate the performance.

II. PROBLEM FORMULATION

The ELD problem is formulated as an optimization problem of minimizing the fuel cost while satisfying several equality and inequality constraints. Usually the network loss is calculated using constant B-loss coefficients, which may lead to sub-optimal solution due to approximations in the computations of these coefficients. More accurate solution can be obtained, if network loss is calculated from load flow. In this paper, Newton-Raphson load flow technique [15] is used to calculate the loss. The constrained optimization problem involving load flow is formulated as follows:

Minimize

$$F_C = \sum_{i=1}^{ng} a_i\, P_{Gi}^{\;2} + b_i\, P_{Gi} + c_i + \left| d_i \sin(e_i (P_{Gi}^{\min} - P_{Gi})) \right| \qquad (1)$$

Subject to:

Real and Reactive power generation limits

$$P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max} \quad : \quad i = 1,2,3 \cdots ,ng \qquad (2)$$

$$Q_{Gi}^{\min} \le Q_{Gi} \le Q_{Gi}^{\max} \quad : \quad i = 1,2,3 \cdots ,ng \qquad (3)$$

Load flow equations

$$P_{Gi} - P_{Di} - p(V,\delta) = 0 \qquad (4)$$

$$Q_{Gi} - Q_{Di} - q(V,\delta) = 0 \qquad (5)$$

III. SELF-ADAPTIVE FIREFLY ALGORITHM

The FA is a Meta heuristic, nature-inspired, optimization algorithm which is based on the social flashing behavior of fireflies, or lighting bugs, in the summer sky in the tropical temperature regions. It was developed by Dr. Xin-She Yang at Cambridge University in 2007, and it is based on the swarm behavior such as fish, insects, or bird schooling in nature. It is similar to other optimization algorithms employing swarm intelligence such as PSO and ABC. But FA is found to have superior performance in many cases [9,10]. FA initially produces a swarm of fireflies located randomly in the search space. The initial distribution is usually produced from a uniform random distribution. The position of each firefly in the search space represents a potential solution of the optimization problem. The dimension of the search space is equal to the number of optimizing parameters in the given problem. The fitness function takes the position of a firefly as input and produces a single numerical output value denoting how good the potential solution is. A fitness value is assigned to each firefly. The FA uses a phenomenon known is bioluminescent communication to model the movement of the fireflies through the search space. The brightness of each firefly depends on the fitness value of that firefly. Each firefly is attracted by the brightness of other fire-flies and tries to move towards them. The velocity or the pull a firefly towards another firefly depends on the attractiveness. The attractiveness depends on the relative distance between the fireflies. It can be a function of the brightness of the fireflies as well. A brighter firefly far away may not be as attractive as a less bright firefly that is closer. In each iterative step, FA computes the brightness and the relative attractiveness of each firefly. Depending on these values, the positions of the fireflies are updated. After a sufficient amount of iterations, all fireflies converge to the best possible position on the search space. The number of fireflies in the swarm is known as the population size, nf . The selection of population size depends on the specific optimization problem. However, typically a population size of 20 to 40 is used for PSO and FA for most applications [9,10]. Each i -th firefly is denoted by a vector x_i as

$$x_i = \left[x_i^1, x_i^2 \cdots, x_i^{nd}\right] \quad (6)$$

The search space is limited by the following inequality

$$x^k(min) \le x^k \le x^k(max) \;:\; k = 1, 2, \cdots, nd \quad (7)$$

Initially, the positions of the fireflies are generated from a uniform distribution using the following equation

$$x_i^k = x^k(min) + \left(x^k(max) - x^k(min)\right) \times rand \quad (8)$$

Here, $rand$ is a random number between 0 and 1, taken from a uniform distribution. Eq. (8) generates random values from a uniform distribution within the prescribed range defined by Eq. (7). The initial distribution does not significantly affect the performance of the algorithm. Each time the algorithm is executed, the optimization process starts with a different set of initial points. However, in each case, the algorithm searches for the optimum solution. In case of multiple possible sets of solutions, the algorithm may converge on different solutions each time. But each of those solutions will be valid as they all will satisfy the requirements.

The light intensity of the i-th firefly, I_i is given by

$$I_i = Fitness\ (x_i) \quad (9)$$

The attractiveness between the i-th and j-th firefly, $\beta_{i,j}$ is given by

$$\beta_{i,j} = \beta_o \exp\left(-\gamma\, r_{i,j}^2\right) \quad (10)$$

Where $r_{i,j}$ is Cartesian distance between i-th and j-th firefly

$$r_{i,j} = \left\|x_i - x_j\right\| = \sqrt{\sum_{k=1}^{nd}\left(x_i^k - x_j^k\right)^2} \quad (11)$$

β_o is a constant taken to be 1. γ is another constant whose value is related to the dynamic range of the solution space. The position of firefly is updated in each iterative step. If the light intensity of j-th firefly is larger than the intensity of the i-th firefly, then the i-th firefly moves towards the j-th firefly and its motion at t-th iteration is denoted by the following equation:

$$x_i(t) = x_i(t-1) + \beta_{i,j}\left(x_j(t-1) - x_i(t-1)\right) + \alpha\left(rand - 0.5\right) \quad (12)$$

α is a constant whose value depends on the dynamic range of the solution space. At each iterative step, the intensity and the attractiveness of each firefly is calculated. The intensity of each firefly is compared with all other fireflies and the positions of the fireflies are updated using (12). After a sufficient number of iterations, all the fireflies converge to the same position in the search space and the global optimum is achieved.

In the above narrated FA, each firefly of the swarm explores the problem space taking into account the results obtained by others, still applying its own randomized moves as well. The influence of other solutions is controlled by the value of attractiveness of Eq. (10), which can be adjusted by modifying two parameters β_o and γ. The first parameter describes attractiveness at $r_{i,j} = 0$ i.e. when two fireflies are found at the same point of solution space. In general $\beta_o \in [0,1]$ should be used and two limiting cases can be defined: The algorithm performs cooperative local search with the brightest firefly strongly determining other fireflies positions, especially in its neighborhood, when β_o = 1 and only non-cooperative distributed random search with β_o = 0.

On the other hand, the value of γ determines the variation of attractiveness with increasing distance from communicated firefly. Setting γ as 0 corresponds to no variation or constant attractiveness and conversely putting γ as ∞ results in attractiveness being close to zero which again is equivalent to the complete random search. In general γ in the range of [0,10] can be chosen for better performance. Indeed, the choice of these parameters affects the final solution and the convergence of the algorithm.

The self-adaptive control of these two parameters during the search process effectively leads the algorithm to land at the global best solution with minimum computational effort. Each firefly with nd decision variables in the FA will be defined to encompass nd +2 FA variables in a self-adaptive method, where the last two variables represent β_o and γ. A firefly can be represented as

$$x_i = \left[x_i^1, x_i^2 \cdots, x_i^{nd}, \beta_{o,i}, \gamma_i\right] \quad (13)$$

Each firefly possessing the solution vector, $\beta_{o,i}$ and γ_i undergo the whole search process of the FA, thereby resulting in better off-springs during the search with lower computational effort. Eq. (10) is accordingly modified as

$$\beta_{i,j} = \beta_{o,i} \exp\left(-\gamma_i\, r_{i,j}^2\right) \quad (14)$$

The self adaptive scheme attempts to prevent sub-optimal solution and enhance the convergence of the algorithm.

IV. PROPOSED ALGORITHM

The proposed SAFA based solution process involves representation of problem variables β_o and γ; and formation of a light intensity function.

IV.1 Representation of decision variables

The decision variables in the PA are real power generation at generator buses except slack bus, β_o and γ. Each firefly in the PA is defined to denote these decision variables in vector form as

$$x = [P_{G2}, \cdots, P_{Gng}, \beta_{o,} \gamma] \tag{15}$$

IV.2 Intensity Function

The SAFA searches for optimal solution by maximizing an intensity function, denoted by I_i, which is formulated from the objective function, Eq. (1).

$$Max \quad I_i = \frac{1}{1 + F_C} \tag{16}$$

It is to be noted that the real power generation, which includes network loss, at slack bus is obtained from the load flow.

IV.3 Stopping Criterion

The process of generating new swarm can be terminated either after a fixed number of iterations or if there is no further significant improvement in the global best solution.

IV.4 Solution Process

An initial swarm of fireflies is obtained by generating random values within their respective limits to every individual in the swarm through Eq. (2). The intensity is calculated by considering the values of each firefly and the movements of fireflies are performed for all the fireflies with a view of maximizing the intensity. The iterative process is continued till convergence. The pseudo code of the PA is as follows.

Read *the Power System Data*
Choose *the number of fireflies in the population,* nf
and $Iter^{\max}$ *for convergence check.*
Generate *the initial population of fireflies*
Set *the iteration counter* $t = 0$
while *(termination requirements are not met) do*
  for $i = 1 : nf$
    Alter the system data and $\beta_o\ \gamma$ *according to* i *-th firefly values*
    Run load flow and obtain slack bus power
    Evaluate F_C *and* I_i *using Eqs. 1 and 16 respectively*
  for $j = 1 : nf$
    Alter the system data according to j *-th firefly values*
    Run load flow and obtain slack bus power
    Evaluate F_C *and* I_j *using Eqs. 1 and 16 respectively*
  if $I_i > I_j$
    Compute r_{ij} *using Eq. (11)*
    Evaluate β_{ij} *using Eq. (14)*
    Move j *-th firefly towards* i *-th firefly through Eq. (12)*
  end*-(if)*
  end*-(* j *)*
  end*-(* i *)*
  Rank the fireflies
end*-(while)*

V. SIMULATIONS

Table 1 FA parameters

Parameter	Value
nf	30
$Iter^{\max}$	300

Table 2 Results of IEEE 14 bus test system

Control Variables (p.u)	Before Optimization	**PA**
P_{G1}	188.974	213.487970
P_{G2}	35	20
P_{G3}	20	15.994770
P_{G6}	12	10
P_{G8}	12	10
β_o	---	0.212654
γ	---	0.827210
Load Demand	259	259
Loss	8.9737	10.4827
Net Fuel	834.6716	826.7192

Cost ($/h)		

The PA is tested on IEEE 14 and 30 bus test systems, whose data have been taken from Ref. [16]. The fuel cost coefficients, lower and upper limits for real power generations for these three test systems are given in Tables A.1-A.2 of the appendix. Programs are developed in Matlab 7.5 and executed on a 2.3 GHz Pentium-IV personal computer. Newton Raphson technique [15] is used to carry out the load flow during the optimization process. The parameters chosen for the PA are given in Table 1.

Table 3 Results of IEEE 30 bus test system

Control Variables (p.u)	Before Optimization	**PA**
P_{G1}	138.539	176.224581
P_{G2}	57.56	48.758822
P_{G5}	24.56	21.478620
P_{G8}	35.0	22.220545
P_{G11}	17.93	12.215761
P_{G13}	16.91	12
β_o	---	0.732605
γ	---	0.253542
Load Demand	283.4	283.4
Loss	7.0990	9.4983
Net Fuel Cost ($/h)	813.6941	802.4630

The optimal solution obtained by the PA for 14 and 30 bus test systems are given along with the initial solution before optimization in Tables 2 and 3 respectively. It is very clear from these tables that the algorithm is able to reduce the fuel cost to the lowest possible values of 826.7192 and 802.4630 $/h from the initial values of 834.6716 and 813.6941 $/h respectively for the three test systems. The resulting loss after optimization is higher for 14 and 30 bus systems, as the objective of the problem is only the minimization of fuel cost in all the test cases. It is to be noted that the loss can further be reduced by adding the loss in the objective function.

VI. SUMMARY

Indeed the FA is a powerful novel population based method for solving complex optimization problems. The convergence and searching capability can be improved with a view to prevent sub-optimal solution through self-adaptive control of FA parameters. In this paper, SAFA solution technique for ELD problem is developed and tested on three IEEE test systems. The algorithm uses NR load flow technique for computing the slack bus power that includes network loss and is able to offer the global best solution at lower computational burden.

ACKNOWLEDGEMENT

The author gratefully acknowledge the authorities of Vaagdevi College of Engineering for their continued support, encouragement and the facilities provided to carry out this work.

APPENDIX

Table A.1 Generator Data for IEEE 14 bus test system

Bus	*a*	*b*	*c*	*d*	*e*	P_{Gi}^{min}	P_{Gi}^{max}
1	0.0016	2.00	150.0	0.0630	50.0	50	300
2	0.0100	2.50	25.0	0.0980	40.0	20	80
3	0.0625	1.00	0.0	0.0	0.0	15	50
6	0.00834	3.25	0.0	0.0	0.0	10	35
8	0.025	3.00	0.0	0.0	0.0	10	30

Table A.2 Generator Data for IEEE 30 bus test system

Bus No	*a*	*b*	*c*	*d*	*e*	P_{Gi}^{min}	P_{Gi}^{max}
1	0.00375	2.00	0	0	0	50	200
2	0.01750	1.75	0	0	0	20	80
5	0.06250	1.00	0	0	0	15	50
8	0.00834	3.25	0	0	0	10	35
11	0.02500	3.00	0	0	0	10	30
13	0.02500	3.00	0	0	0	12	40

REFERENCES

1. Wood. AJ and Woolenberg. BF. (1996). Power generation, operation and control, John Willey and Sons, New York
2. Chowdhury.BH and Rahman. S. (1990). A review of recent advances in economic dispatch, IEEE Trans. on Power Systems, 5(4), 1248-1259.
3. Panigrahi. CK, Chattopadhyah. PK, Chakrabarti. RN and Basu. N. (2006). Simulated annealing technique for dynamic economic dispatch, Electric Power Components and Systems, 34(5), 577-586.
4. Adhinarayanan. T and Sydulu. M. (2008). Diretional search genetic algorithm applications to economic dispatch of thermal units, International Journal for Computational Methods in Engineering Science and Mechanics, 9(4), 211-216.
5. He Da-Kuo, Wang Fu-li and Mao Zhi-zhong. (2008). Hybrid genetic algorithm for economic dispatch with value-point effect, Electric Power Systems Research, 78, 626-633.
6. JB. Park, KS. Lee, JR. Shin and KY.Lee. (2005). A particle swarm optimization for economic dispatch with nonsmooth cost function, IEEE Trans Power Syst, 20(1): 34-42.
7. Subbaraj. P, Rengaraj. R, Salivahanan. S and Senthilkumar. TR. (2010) Parallel particle swarm optimisation with modified stochastic acceleration factors for solving large scale economic dispatch problem, Electrical Power and Energy Systems, 32(9), 1014-1023.
8. A.Pereira-Neto, C.Unsihuay and OR.Saavedra. (2005). Efficient evolutionary strategy optimization procedure to solve the nonconvex economic dispatch problem with generator constraints. IEEE Proc Gener Transm Distrib., 152(5): 653-660.
9. X. S. Yang, Nature-Inspired Meta-Heuristic Algorithms, Luniver Press, Beckington, UK, 2008.
10. X. S. Yang. (2009). Firefly algorithms for multimodal optimization, in Proceedings of the Stochastic Algorithms: Foundations and Applications (SAGA '09), Vol. 5792 of Lecture Notes in Computing Sciences, pp. 178–178, Springer, Sapporo, Japan.
11. Kuldeep Kumar Swarnkar. (2012). Economic load dispatch problem with reduce power loss using firefly algorithm, Journal of Advanced Computer Science and Technology, 1(2): 42-56.

12. K. Vinod Kumar and G. Lakshmi Phani. (2011). Combined economic emission dispatch-pareto optimal front approach, International journal of computer applications, 30(12): 16-21.
13. M. H. Sulaiman, M. W. Mustafa, Z. N. Zakaria, O. Aliman and S. R. Abdul Rahim. Firefly Algorithm Technique for Solving Economic Dispatch Problem, 2012 IEEE International Power Engineering and Optimization Conference (PEOCO2012), Melaka, Malaysia: 6-7 June 2012.
14. K.Chandrasekaran and Sishaj P Simon. (2012). Tuned Fuzzy Adapted Firefly Lambda Algorithm for Solving Unit Commitment Problem, J. Electrical Systems,8(2): 132-150.
15. W.F. Tinney and C.E. Hart. (1967). Power flow solution by Newton's method, IEEE Trans. PAS-86: 1449-1460.
16. Test Systems Archive, Available at http://www.ee.washington.edu/research/pstca/ (Accessed December 2012).