Advanced Research in Electrical and Electronic Engineering
Print ISSN: 2349-5804; Online ISSN: 2349-5812 Volume 2, Number 3 January-March (2015) pp. 205-209

http://www.krishisanskriti.org/areee.html

Comparison of Results among Three Optimization Algorithms Applied to an Economic Load Dispatch Problem

G. Loganathan[1] and D. Rajkumar[2]

[1]PG Scholar, Department of EEE M. Kumarasamy College of Engineering Karur – 639001, Tamilnadu, India
[2]Department of EEE M. Kumarasamy College of Engineering Karur – 639001, Tamilnadu, India
E-mail: [1]loganathankarur@gmail.com, [2]rajgdr@gmail.com

Abstract—*In India nowadays meeting the power demand became a huge task for the government. Increase in generation cost and power demand requires an economic operation for the generators and consumers. Economic Dispatch plays a vital role in power system generation, operation and control. The proposed time effective new algorithm called as Shuffled Frog Leaping algorithm deals with the economic dispatch for the optimal allocation of generators and reduction of generation fuel cost in the thermal power plants. In this technique, constraints such as power balance and generation capacity are taken care. The feasibility of this proposed method is demonstrated in three units and six units system and compared with other naturally inspired algorithms. The algorithm has the better convergence rate and better performance in different number of cases. With the quality of results obtained, this proposed method can be used to perform better operation of an economic dispatch in the practical power system.*

Index Terms*: Economic Dispatch (ED), Shuffled Frog Leaping Algorithm (SFLA), Particle Swarm Optimization (PSO)*

1. INTRODUCTION

Electricity demand in India is growing at rapid speed. The present Peak Demand is about 132507 MW and the Installed Capacity is 1, 28, 083 MW with generation mix of 63% thermal, 25% hydro, 9% Nuclear and 9% renewable. The projected Peak Demand in 2017 is more than 200 GW. The corresponding installed capacity requirement in 2017 is more than 300 GW. Thus, the demand can be managed by minimizing the losses, simultaneous increase in generation and the most vital economic operations. Economic operation, planning and control of electrical power systems have always been a vital concern in the electrical power industry. The efficient use of the available fuel which comes mostly from irreplaceable natural resources results in optimal economic operation of power generation systems. A crucial truth that increases the importance of the optimal economic operations is that the electrical energy cannot be stored in large amounts. Considering the efficiency of thermal systems (usually 50-60%), even a small significant reduction in the amount of fuel results in small percent of savings which is an advantageous one in the interest of our country. The dissimilarity of different generating units due to various reasons such as their characteristics, efficiencies and the distances between their locations and load centre's results in quite different operating costs. As a result, an optimal power generation schedule that determines the generation level of each of the units is vital to meet the load demand at the minimum cost. In addition, the operating cost of a specific generating unit is not linearly dependent on the power it produces. The optimal economic generation schedule can only be realized by considering various operational constraints and limitations. The load demand must be satisfied all the time while including the system losses that are function of the power generation. Suitable improvements in the unit outputs scheduling can contribute to significant cost savings.

The most modern control centres which are installed today uses the classical methods to solve a well known exact co-ordination equations [12]. The difference between different approaches that are being used is the method used to solve the co- ordinations equations. The co-ordination equations are generally solved by interactively adjusting the load until the sum of the generator output matches the system load, system loss which simultaneously should result in minimum cost of power generation. The classical methods are complex and also time consuming [14]. These problems are eliminated in the naturally inspired algorithms like ACO, GA and PSO [13]. Truly speaking, each and every method has its own merits and demerits. SFLA is a global search technique originally introduced by Eusuff and Lansey. This concept was obtained from the concept of group of frogs. The most attractive features of SFLA are: simple concept, easy implementation, fast computation, robust search ability and robustness in controlling the parameters [15][16]. The SFLA has the capability of quick convergence[10].

There are many attempts taken to overcome the disadvantages in any method. Based on the experience, when compared with other conventional methods like lambda iteration method, gradient method the SFLA is very fast in computing and simultaneously gives accurate solutions. Compared with the other algorithms, SFLA shows unique advantages in searching speed and accuracy. In brief, the SFLA is regarded as a simple heuristic of well-balanced mechanism with flexibility to enhance and adapt to both global and local exploration abilities by changing the value of inertia weight, gains the attention of researchers to apply this concept to power system applications. SFLA is better in global search ability.

2. ECONOMIC LOAD DISPATCH PROBLEM

The main use of economic dispatch problem is to reduce the fuel cost of the generating units of the thermal power plant subjected to the operating constraints of the power system. The objective function of the economic load dispatch problem is formulated by

$$MinF_T = \sum_{i=1}^{N_g} F_i(P_{Gi})$$

The operating equality and inequality constraints are formulated as

$$Pgimin \leq Pgi \leq Pgimax, i = 1, 2, \ldots\ldots, Ng$$

Where,

F_T – Total fuel cost to be minimized

$F_i(P_{Gi})$– Fuel cost of i^{th} generating unit

N_g – Number of thermal generating units

P_{Gi} – Power generated from n^{th} generator

P_D – Total Power demand

P_{gi}^{min} – minimum power generation for unit i

P_{gi}^{max} – maximum power generation for unit i

The fuel cost of a generator can be calculated by

$$Fi(Pi) = ai\, Pi + bi\, Pi + ci$$

$a_{i,}$ b_i , c_i are the cost coefficients of the i^{th} unit

3. SHUFFLED FROG LEAPING ALGORITHM

The SFLA is a meta- heuristic optimisation method which combines the advantages of both the genetic based MA algorithm and social behaviour based PSO algorithm. This method is based on the imitating, observing and modelling the behaviour of group of frogs when searching for the food location which has maximum amount of food. The SFLA is originally introduced by Eusuff and Lansey in 2003. It can be used for solving various complex optimisation problems.

This method can be applied in all engineering disciplines such as Travelling Salesman Problem (TSP), Water Resource Distribution and job-shop scheduling arrangement. The main benefit of this algorithm is its faster convergence speed than other optimisation techniques. In this SFLA, the frogs are separated into groups which are termed as memeplexes. In each memeplexes there are many numbers of frogs and each frogs has its individual ideas within the memplexes. There are many memeplexes and in each has different frogs with ideas and ideas of each frogs can be infected by other frogs in same memeplex. After a defined number of memetic evolution steps, ideas of each frogs between memplexes are passed in a shuffling process. The local search and the shuffling process continued until the convergence criteria are satisfied.

The flowchart of SFLA is illustrated in the Fig. .1. In this first step of this method, an initial population of P frogs is generated randomly within the feasible search space. The position of the i^{th} frog is represented as $Xi = (Xi1, Xi2, \ldots, XiD)$, where D is the number of variables. Then the fitness (F) of the each frog is identified and the frogs are sorted in descending order based on their fitness. Then the entire population is portioned into an m subsets referred to as memeplexes, each containing n frogs

($P= m * n$). The portioning method is as follows: the first frog goes into the first memeplex, the second frog goes to the second memeplex, the m^{th} frog goes into the mth memplex, and (m+1) th frog goes into the first memeplex and so on. In all memeplexes the fitness is calculated for each frog with their positions. Both the best fitness and worst fitness of each frogs are calculated and identified as Xb and Xw respectively.

Also the position of a frog with the best global fitness is identified and represented as Xg. Then within the each memeplex, a process similar to the Particle Swarm Optimisation (PSO) algorithm is applied to improve the fitness of the worst frog. The worst fitness frog is improved to an best fitness frog with best position and is calculated by

$$D = rand * (Xg – Xw) \qquad \textbf{----(1)}$$

$$X = Xw + D (D\, min < D < D\, max) \qquad \textbf{----(2)}$$

Where, D max and D min are the maximum and minimum step sizes allowed for each frogs position respectively.

This process gives the better solution and so it will replace the worst frog. Otherwise, the calculations in (1) and (2) are repeated but *Xb* is replaced by *Xg*. If there is no improvement in this case, a new solution will be randomly generated within the feasible space to replace it. The calculations will continue for a specific number of iterations [10]. Therefore, SFLA simultaneously performs an independent local search in each memeplex using a process similar to the PSO algorithm. The flowchart of local search of SFLA is illustrated in Fig. 1.

After a predefined number of memetic evolutionary steps within each memeplex, the solutions of evolved memeplexes *(X1, ……, XP)* are replaced into new population (new population = {Xk, k = 1, …, P}); this is called the shuffling process. The shuffling process promotes a global information exchange among the frogs. Then, the population is sorted in order of decreasing performance value and updates the population best frog's position Xg, repartition the frog group into memeplexes, and progress the evolution within each memeplex until the conversion criteria are satisfied. Usually, the convergence criteria can be defined as follows [17].

1)The relative change in the fitness of the global frog within a number of consecutive shuffling iterations is less than a pre-specified tolerance. 2)The maximum predefined number of shuffling iteration has been obtained.

4. IMPLEMENTATION OF SFLA

1) Population size: It denotes the total number of frogs. In this simulation procedure, population size has been taken as 100. Increase in number of population means good accuracy but it will lead to more propagation delay. After running the program with different number of population size, it has been observed that for this optimization problem, typically a population size of 100 is most suited for optimizing both processing time and value.
2) Number of memplexes: In this programming, number of memplexes is taken as 10. As population size and number of memplexes are user input, the given input of number of memplexes is such that there exists a certain number of frogs (population size/total number. of memplexes) in each memplexes.
3) Number of global iteration: In this type of iteration, the cross-over between best frog & worst frog is done taking the whole population. One global iteration consists of local iterations as many as number of memplexes present. It is taken 10 here. Maximizing the number of global iteration gives more accurate results but it takes more time to process.
4) Number of local iteration: In this type of iteration, the cross-over between best frog & worst frog is done in every single memplexes. Number of local iterations are taken as 20 here. Maximizing the number of local iterations also gives more accuracy but it gives more delay.

All the SFLA parameters value discussed above is for three units and six units test system. shuffled frog leaping algorithm is as follows

Step 1: Initialisation of power demand and constraints.

Step2: An initial population of P frogs are created randomly for an S-dimensional problem.

Step3: Determine fitness value of each frog according to the given problem. Then Record the best frog position in the entire population.

Step4: The frogs are arranged in a descending order according to their fitness.

Step5: The P frogs are partitioned into m memeplexes, each containing n frogs (P = m × n).

Step 6: In each memeplex, the frogs with the best and the worst fitness are determined and named as Xb and Xw, respectively. Also, the position of frog with the global best fitness among the memeplexes is identified as Xg

Step7: After a defined number of memeplex evolution steps, all frogs of memeplexes are collected, and sorted in descending order based on their fitness. Step 6 divides frogs into different memeplexes again and then step 6 is performed.

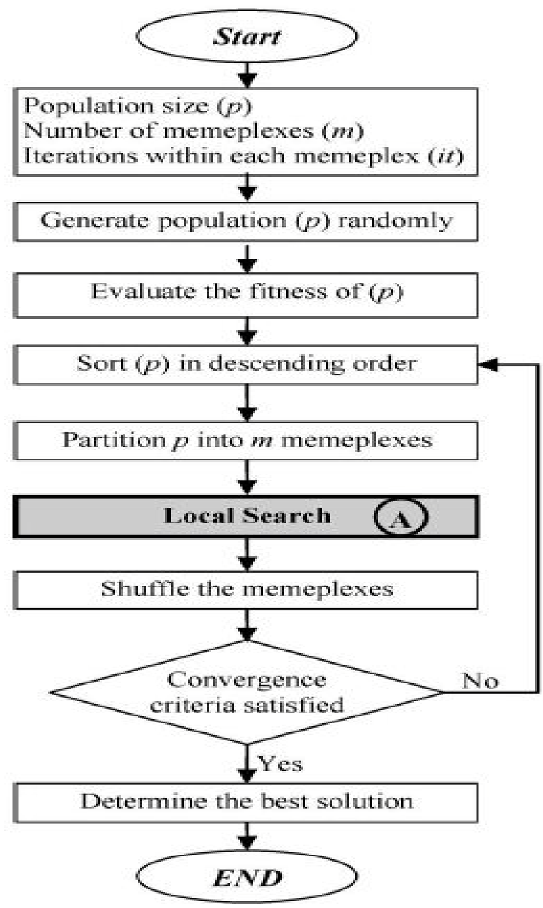

Fig. 1. Global search in SFLA

Step8: If the defined convergence criteria are satisfied or the output does not change for a specific number of iterations, the program will be terminated and the results will be printed, and the rest of the program goes to Step 5

Step 9: end the procedure

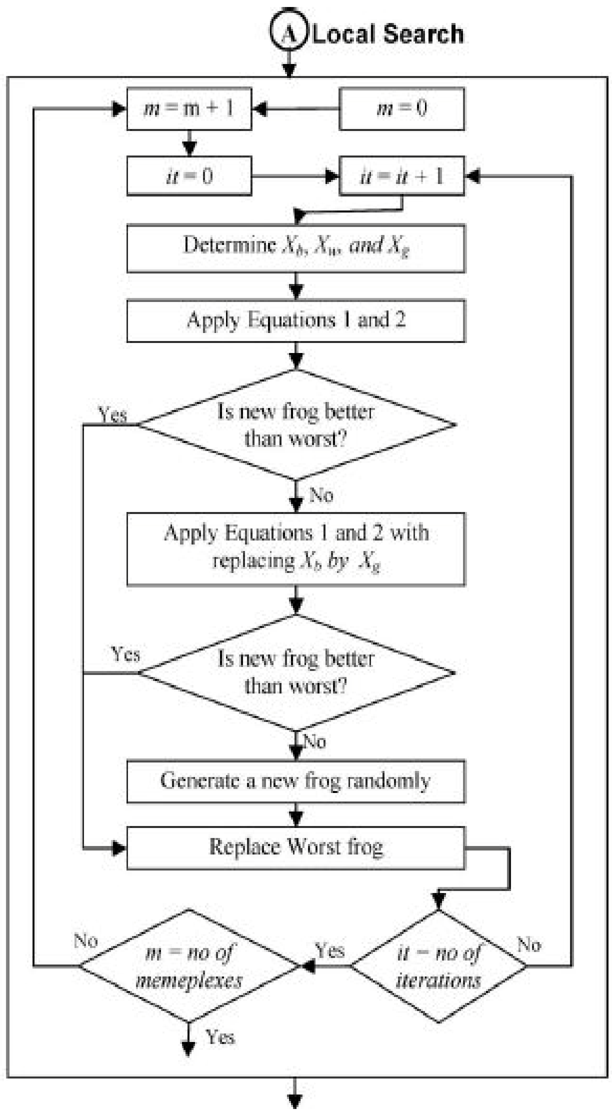

Fig. 2: Local search in SFLA

5. EXPERIMENTAL RESULTS

The Proposed SFLA Algorithm has been applied to ED problems in two different thermal unit systems for verifying its feasibility. These are a three units system and a six units system [4] [18]. The transmission losses will not take into account in all the case studies here for the sake of comparison with other algorithms presented in literature [4] [5] [9] [19] . The stopping criterion, maximum number of iteration, varies for each case in considering the problem scale. The software has been written in MATLAB language and executed in Pentium ® Core 2 Duo personal computer with 2GB RAM.

5.1 Case study 1- Three units system

In this example, a simple system with three thermal units is used to demonstrate how the proposed approach works. The load demand is 850MW.The unit characteristics are given in Table 1. In this case, each individual Pg contains three generator power outputs, such as P1, P2, and P3, which are generated randomly.Now, Table 2 provides the statistic results that involved the generation cost, evaluation value, and average execution time.

Table 1 : Generating Unit's Capacity and Coefficients

Unit	Pmin MW	Pmax MW	ai $	bi $/MW	ci $/MW2
1	100	600	0.00156	7.92	561
2	100	400	0.00194	7.85	310
3	50	200	0.00482	7.97	78

Table 2: Resluts For 3-Generator System

Unit Output	PSO	SFLA	GA
P1 (MW)	393.1698	393.0192	393.0103
P2 (MW)	334.6037	335.2319	319.2256
P3 (MW)	122.2264	121.7489	137.7643
Total Power Output (MW)	850	850	850
Total Generation Cost ($/h)	8194.3561	8194.04909	8194.9790
Execution time (sec)	1.777847	0.146978	1.82316

5.2 Case study 2- Six units system

The system contains six thermal units, 26 buses, and 46 transmission lines [10]. The load demand is 1800MW. The characteristics of the six thermal units are given in Tables 3. In this case, each individual Pg contains six generator power outputs, such as P1, P2, P3, P4, P5 and P6, which are generated randomly. Now, Table 4 provides the statistic results that involved the generation cost, evaluation value, and average execution time.

Table 3: Generating Unit's Capacity and Coefficients

Unit	Pmin MW	Pmax MW	ai $	bi $/MW	ci $/MW2
1	**150**	**600**	**0.002035**	**8.43205**	**85.6438**
2	**150**	**600**	**0.003866**	**6.41031**	**303.7780**
3	**150**	**600**	**0.002182**	**7.42890**	**847.1484**
4	**150**	**600**	**0.001345**	**8.30154**	**274.2241**
5	**150**	**600**	**0.002182**	**7.42890**	**847.1484**
6	**150**	**600**	**0.005963**	**6.91559**	**202.0258**

Table 4: Results For 6-Generator System

Unit Output	PSO	SFLA	GA
P1 (MW)	177.4313	174.6187	176.5212
P2 (MW)	361.5820	358.5631	355.4563
P3 (MW)	383.5298	386.5549	387.3982
P4 (MW)	320.9923	319.9955	321.2361
P5 (MW)	381.6587	377.7851	378.9237
P6 (MW)	174.7429	182.4827	180.4645
Total Power Output (MW)	1800	1800	1800

Total Generation Cost ($/h)	17464.68325	17460.33748	17467.22863
Execution time (sec)	0.691814	0.150432	0.728479

6. CONCLUSION

The proposed shuffled frog leaping algorithm is implemented successfully for the economic dispatch problem in practical power system considering constraints and transmission losses. This proposed method has been applied to three units and six units system. This algorithm gives a result with best convergence characteristics and better quality of solution when compared with other methods such as PSO and GA. The computation time and convergence characteristics are very fast than other methods like PSO and GA.

REFERENCES

[1] A. J. Wood and B. F. Wollenberg, Power generation operation and control, 2nd ed. John Willy and Sons, 1996.

[2] B. H. Chowdhury and S. Rahman, “A review of recent advances in economic dispatch, ” IEEE Trans. Power Systems, vol. 5, no. 4, pp. 1248–1259, Nov. 1990.

[3] P. H. Chen and H. C. Chang, “Large Scale Economic Dispatch by Genetic Algorithm”, IEEE Trans. on Power Systems, vol. 10, no. 4, Nov. 1995.

[4] Z. X. Liang and J. D. Glover, “A zoom feature for a dynamic programming solution to economic dispatch including transmission losses”, IEEE Trans. on Power Systems, vol. 7, no. 2, pp. 544-550, May 1992.

[5] M.Vanitha and K. Thanushkodi, “Solution to Economic Dispatch Problem by Differential Evolution Algorithm Considering Linear Equality and Inequality Constrains, ” IJRRECE, vol. 1, no. 1, March 2011.

[6] D. N. Simopoulos, S. D. Kavatza and C. D. Vournas, “Unit commitment by an enhanced simulated annealing algorithm, ” IEEE Trans. on Power Systems, vol. 21, No. 1, Feb. 2006.

[7] X.-H. Luo, Y. Yang, and X. Li, “Solving TSP with shuffled frog-leaping algorithm, ” in Proc. ISDA 2008, vol. 3, pp. 228–232.

[8] A. Rahimi-Vahed and A. H. Mirzaei, “Solving a bi-criteria permutation flow-shop problem using shuffled frog-leaping algorithm, ” in Soft Computing. New York: Springer-Verlag, 2007.

[9] M. M. Eusuff, K. E. Lansey, and F. Pasha, “Shuffled frog-leaping algorithm: A memetic meta-heuristic for discrete optimization, ” Eng. Optimiz., vol. 38, no. 2, pp. 129–154, 2006.

[10] H. Elbehairy, E. Elbeltagi, and T. Hegazy, “Comparison of two evolutionary algorithms for optimization of bridge deck repairs, ” Computer- Aided Civil Infrastr. Eng., vol. 21, pp. 561–572, 2006.

[11] D. N. Simopoulos, S. D. Kavatza and C. D. Vournas, “Unit commitment by an enhanced simulated annealing algorithm, ” IEEE Trans. on Power Systems, vol. 21, No. 1, Feb. 2006.

[12] Y. M. Park, J. R. Won, and J. B. Park, “A new approach to economic load dispatch based on improved evolutionary programming, ” *Eng. Intell.Syst. Elect. Eng. Commun.*, vol. 6, no. 2, pp. 103–110, June 1998.

[13] N. Yorino, H. M. Hafiz, Y. Sasaki, and Y. Zoka, “High-speed real-time dynamic economic load dispatch, ” IEEE Trans. Power Syst., vol. 27, no. 2, pp. 621–630, May 2012.

[14] K. Y. Lee and M. A. El-Sharkawi, Eds., Modern Heuristic Optimization Techniques with Applications to *Power Systems*: IEEE Power Engineering Society (02TP160), 2002.

[15] M. Renu Avinaash, G. Ravi Kumar, K. Anjaneya Bhargav, T. Srikanth Prabhu and D. IndraSena Reddi, ” Simulated Annealing Approach to Solution of Multi-Objective Optimal Economic Dispatch, ” IEEE 2012.

[16] A. Rahimi-Vahed and A. H. Mirzaei, “Solving a bi-criteria permutation flow-shop problem using shuffled frog-leaping algorithm, ” in Soft Computing. New York: Springer-Verlag, 2007.

[17] T. H. Huynh, “A modified shuffled frog leaping algorithm for optimal tuning of multivariable PID controllers, ” in Proc. ICIT 2008, pp. 1–6.

[18] Z. L. Gaing, “Particle Swarm Optimization to Solving the Economic Dispatch Considering the Generator Constraints”, IEEE Trans. on Power Systems, vol. 18, No. 3, Aug. 2003.

[19] A. Bakirtzis, V. Petridis, and S. Kazarlis, “Genetic algorithm solution to the economic dispatch problem, ” Proc. Inst. Elect. Eng.–Gen., Transm. Dist., vol. 141, no. 4, pp. 377–382, July 1994.