APPLICATION OF GENETIC ALGORITHM IN SOLVING ECONOMIC DISPATCH WITH MULTIPLE FUEL OPTIONS

[1]P. K. Olulope, [2]O. M. Amusan, [3]Bankole Adebanji,

[1,2,3]Department of Electrical and Electronic Engineering, Ekiti State University Ado-Ekiti, Nigeria

IJASR 2021
VOLUME 4
ISSUE 3 MAY – JUNE **ISSN: 2581-7876**

Abstract: Minimizing electricity generation cost (including fuel cost, plus emission cost, plus operation/maintenance cost, plus network loss cost) of multiple operating units has been a major issue in the power sector. The economic dispatch has the objective of allocating different loads to the power generators in such a manner that the total fuel cost is minimized while all operating constraints are satisfied. Conventional optimization methods assume generator cost curves to be continuous and monotonically increasing, but modern generators have a variety of nonlinearities in their cost curves making this assumption inaccurate, and the resulting approximate dispatches cause a lot of revenue loss. Evolutionary methods like Genetic Algorithm perform better for such problems. To know the effectiveness and efficiency Genetic Algorithm in solving economic dispatch, this paper proposes the application of these evolutionary algorithms to three-generators and six-generator.

Firstly, the mathematical model of economic dispatch is developed and then, the Genetic Algorithm is developed to solve the economic dispatch problem. The complex problem of economic power dispatch is solved using genetic algorithm. The test results clearly demonstrated that genetic algorithm which is capable of achieving global solutions is simple, good computational efficiency and has quite a stable dynamic convergence characteristic. In the case economic dispatch problem with multiple fuel option, the genetic algorithm has shown capability of achieving result as the conventional numerical method.

Keywords: Economic Dispatch, Genetic Algorithm, Generation-Demand, Operation cost, maintenance cost, transmission losses, Heuristic search algorithm

1. INTRODUCTION

Economic dispatch is one of the major optimization issues in power system. Its objective is to allocate the power demand among committed generators in the most economical manner, while all physical and operational constraints are satisfied. The cost of power generation, particularly in fossil fuel plants, is very high and economic dispatch helps in saving a significant amount of revenue. Conventional methods like lamda iteration, quadratically constrained programming, gradient methods etc. rely heavily on the convexity assumption of generator cost curves and usually approximate these curves using quadratic or piecewise quadratic monotonically increasing cost functions [1]. This assumption is not valid because the cost functions of modern generators have discontinuities and higher order nonlinearities due to valve point loading [2], [3], prohibited operating zones and ramp rate limits of generators [4]. The practical economic dispatch with above nonlinearities translates into a complicated optimization problem having complex and non-convex characteristics, with multiple minima, making the challenge of obtaining the global minima, very difficult. Conventional gradient based optimization methods fail to model these discontinuities and usually result in inaccurate dispatches causing loss of revenue. To achieve a fast and near global optimal solution, methods like particle swarm optimization and genetic algorithm are often used.

Genetic Algorithms (GAs) are adaptive heuristic search algorithm based on the evolutionary ideas of natural selection and genetics. As such they represent an intelligent exploitation of a random search used to solve optimization problems. Although randomized, genetic algorithms are by no means random, instead they exploit historical information to direct the search into the region of better performance within the search space. [5]

The Genetic Algorithm was introduced in the mid-1970s by John Holland and his colleagues and students at the University of Michigan. The GA employs the principle first laid down by Charles Darwin of "survival of the fittest" in its search process to select and generate individuals (design solutions) that are adapted to their environment (design objectives/constraints). Therefore, over a number of generations (iterations), desirable traits (design characteristics) will evolve and remain in the genome composition of the population (set of design solutions generated each iteration) over traits with weaker undesirable characteristics. The genetic algorithm is well suited to and has been extensively applied to solve complex design optimization problems because it can handle both discrete and continuous variables and nonlinear objective and constrain functions without requiring gradient information. [6]

However in practical, generating units utilizes more than one fuel such as coal, gas, Oil, nuclear or water which limits the power plant operation. In such cases the economic load dispatch problem should be carried out in different way. There arises a concept of economic load dispatch with multiple fuel options. The cost function of a generator is much nonlinear and with more discontinuous [6].

2. LITERATURE REVIEW

Review of Economic Dispatch

Economic Dispatch (ED) is defined as the process of allocating generation levels to the generating units in the mix, so that the system load is supplied entirely and most economically. In static economic dispatch, the objective of the conventional economic dispatch problem is to minimize the total cost of thermal generating units while satisfying various constraints including power balance and generator power limits. In the economic dispatch problem with multiple fuel options, the piecewise quadratic function is used to represent the multiple fuels which are available for each generating units [3]. There have been many algorithms proposed for economic dispatch. These include:

1. Merit Order Loading
2. Range Elimination
3. Binary Section
4. Secant Section
5. Graphical/Table Look-Up
6. Convex Simplex
7. Dantzig-Wolf Decomposition etc.[7]

The following are well-known examples of “intelligent” algorithms that use clever simplifications and methods to solve computationally complex problems.

1. Swarm Intelligence
2. Tabu Search
3. Simulated Annealing
4. Genetic Algorithms
5. Artificial Neural Networks
6. Support Vector Machines[8]

Review of Genetic Algorithm

Genetic algorithm is a stochastic global search method [9]. Genetic algorithm uses the objective function of the problem without demanding its differentiability and linearity. The search process starts through a population of points not with a single point. The three basic pillars of genetic algorithm are Selection, Crossover and Mutation operations. Initially chromosomes are selected to be parents randomly, which may not lead to global solution. The selection parameter selects the chromosomes which has high fitness value. There are many methods how to select the best chromosomes, for example Roulette wheel selection, rank selection, tournament selection and some others. Crossover is a process of taking two parents randomly and produces a new offspring solution. [10]

The idea behind the crossover is the new chromosome is better than their parents. This is analogous to reproduction. There are many methods for crossover, for example single point, two points and uniform crossover. The Mutation operator is to maintain diversity of solutions and to enlarge the information contained in the population. This is a random process where one allele of gene is replaced by another to produce the new genetic structure. [10]

The following steps lead to the computation of genetic algorithm

Representation

Genetic Algorithms are derived from a study of biological systems. In biological systems evolution takes place on organic devices used to encode the structure of living beings. These organic devices are known as chromosomes. A living being is only a decoded structure of the chromosomes. Natural selection is the link between chromosomes and the performance of their decoded structures. In GA, the design variables or features that characterize an individual are represented in an ordered list called a string. Each design variable corresponds to a gene and the string of genes corresponds to a chromosome. Chromosomes are made of discrete units called genes. [11]

Encoding:

Normally, a chromosome corresponds to a unique solution **x** in the solution space. This requires a mapping mechanism between the solution space and the chromosomes. This mapping is called an encoding. In fact, Genetic algorithm works on the *encoding* of a problem, not on the problem itself. The application of a genetic algorithm to a problem starts with the encoding. The encoding specifies a mapping that transforms a possible solution to the problem into a structure containing a collection of decision variables that are relevant to the problem. [11]

Decoding:

Decoding is the process of conversion of the binary structure of the chromosomes into decimal equivalents of the feature values. Usually, this process is done after de-catenation of the entire chromosome to individual chromosomes. The decoded feature values are used to compute the problem characteristics like the objective function, fitness values, constraint violation and system statistical characteristics like variance, standard deviation and rate of convergence. The stages of selection, crossover, mutation etc are repeated till some termination condition is reached. The equivalent decimal integer of binary string λ is obtained as

$$y^j \sum_{i=1}^{i} 2^{i-1} b_i^j \quad (j = 1,2, \ldots..n) \qquad (1)$$

Where b_i^j is the i^{th}binary digit of the j^{th}string, $i$is the length of the string, n is the number of strings or populationsize.

The continuous variable λcan be obtained to represent a point in the search space according to a fixed mapping rule, i.e

.

$$\lambda^j = \lambda^{min} + \frac{\lambda^{max} + \lambda^{min}}{2^{i}-1} y^j \; (j = \; 1,2, \ldots \ldots, n) \qquad (2)$$

Where λ^{min} is the minimum number of variable, λ^{max} is the maximum value of variable, $y\,j$ is the binary coded value of the string. [11]

Initialization:

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions (the *search space*). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found. [11]

Evaluation:

Suitability of the solutions is determined from the initial set of solution of the problem. For this suitability determination, we use a function called fitness function. This function is derived from the objective function and used in successive genetic operation. The evaluation function is a procedure for establishing the fitness of each chromosome in the population and is very much application orientated. [11]

Since Genetic algorithms proceed in the direction of evolving the fittest chromosomes, and the performance is highly sensitive to the fitness values. In the case of optimization routines, the fitness is the value of the objective function to be optimized. Penalty functions can also be incorporated into the objective function, in order to achieve a constrained problem.

Fitness Function:

The Genetic algorithm is based on Darwin's principle that "The candidates, which can survive, will live, others would die". This principal is used to find fitness value of the process for solving maximization problems. Minimization problems are usually transferred into maximization problems using some suitable transformations. Fitness value $f(x)$ is derived from the objective function and is used in successive genetic operations. The fitness function for maximization problem can be used the same as objective function $F(X)$ [11-15].

The fitness function for the maximization problem is:

$$f(x) = F(X) \quad (3)$$

For minimization problems, the fitness function is an equivalent maximization problem chosen such that the optimum point remains unchanged. The following fitness function is often used in minimization problems:

$$F(X) = 1/(1 + f(x)) \quad (4)$$

Here $f(x)$ is fitness function and $F(X)$ is objective function.

Selection:

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a *fitness-based* process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and prefere5ntially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time consuming.

Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include roulette wheel selection and tournament selection. [11]

Reproduction:

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation. For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired," some research suggests more than two "parents" are better to be used to reproduce a good quality chromosome.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above. [11].

Termination:

This generational process is repeated until a termination condition has been reached. [12] Common terminating conditions are:

1.) A solution is found that satisfies minimum criteria
2.) Fixed number of generations reached
3.) Allocated budget (computation time/money) reached
4.) The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results

5.) Manual inspection
6.) Combinations of the above.

Flow Chart of Genetic Algorithm

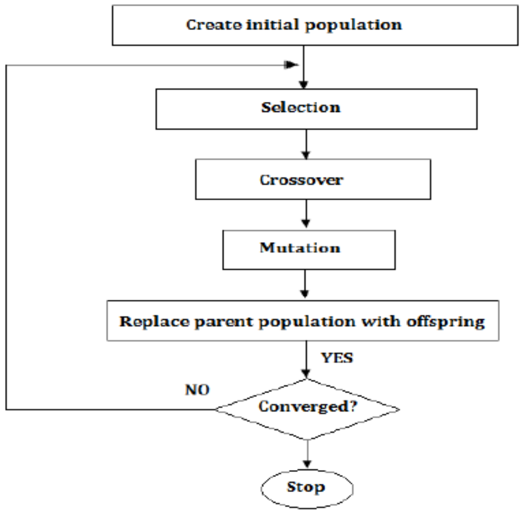

Figure 1: Flow chart of Genetic Algorithm [11]

Implementation genetic algorithm to Economic dispatch with multiple fuel options Problem [9]
Step 1: Read all the data of the system, and initialize the population (Real powers) randomly.
Step 2: Check for Power balance constraint and power limit constraints. Form a pooling mate which satisfies the above two conditions.
Step 3: Calculate the fitness value for every single chromosome.
Step 4: By using Roulette wheel selection procedure select those strings which have highest fitness value.
Step 5: Apply genetic algorithm operator's crossover and mutation.
Step 6: Check for terminating condition. If the termination condition was satisfied stop the generations and print the cost else update the pooling mate and go to the Step 3.

3. METHODOLOGY

Modelling of Economic Dispatch and Problem Formulation

The objective of the conventional economic dispatch problem is to minimize the total cost of thermal generating units while satisfying various constraints including power balance and generator power limits. In the economic dispatch problem with multiple fuel options, the piecewise quadratic function is used to represent the multiple fuels which are available for each generating units [3]. Therefore, the objective of the economic dispatch problem with multiple fuel options is to find a suitable fuel for each generating unit so as their total cost is minimized while satisfying different constraints including power balance and generation limits.

Mathematically, the problem is formulated as follows:

$$Min\ F = \sum_{i=1}^{n} F_i(P_i) \qquad (5)$$

In general, a piecewise quadratic function is used to represent the input-output curve of a generator with multiple fuels [13] and described as

$$F_i(P_i) = \begin{cases} a_{i1} + b_{i1}P_i + c_{i1}P_i^2, \ Fuel\ 1\ P_{imin} \leq P_i \leq P_{i1} \\ a_{i2} + b_{i2}P_i + c_{i2}P_i^2, \ Fuel\ 2 \quad P_{i1} \leq P_i \leq P_{i2} \\ \cdot \\ \cdot \\ \cdot \\ a_{ik} + b_{ik}P_i + c_{ik}P_i^2, \ Fuel\ k\ P_{ik-1} \leq P_i \leq P_{imax} \end{cases} \tag{6}$$

Where

a_{ik}, b_{ik}, c_{ik}	Cost Coefficient for unit *i* for fuel type *k*
P_i	Output power of unit *i* (MW)
P_{imin}, P_{imax}	Lower and Upper generation limits of unit i

Subject to

a) Power balance constraint

$$\sum_{i=1}^{n} P_i - P_L - P_D = 0 \tag{7}$$

Where the power loss is approximately calculated by Kron's formula:

$$P_L = \sum_{i=1}^{N} \sum_{i=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{0i} P_i + B_{00} \tag{8}$$

b) Generator operating limits:

$$P_{i,min} \leq P_i \leq P_{i,max};\ i = 1, \ldots, N \tag{9}$$

Where

P_i	Output power of unit *i*
P_D	Total load demand of the system (MW)
P_L	Total network loss of the system (MW)
B_{ij}, B_{oi}, B_{oo}	Transmission loss formula coefficients

Table I: Data for 3-Unit System

S/N	Generating Units	Lower Limit, P_{min} (MW)	Upper Limit, P_{max} (MW)	Cost Coefficient (a, b, c)
1	Generator 1	10	85	0.008, 7, 200
2	Generator 2	10	80	0.009, 6.3, 180
3	Generator 3	10	70	0.007, 6.8, 140

$$B_{ij} = \begin{matrix} 0.000218 & 0.000093 & 0.000028 \\ 0.000093 & 0.000228 & 0.000017 \\ 0.000028 & 0.000031 & 0.000015 \end{matrix}$$

$$B_{0i} = 0.0003 \quad 0.0031 \quad 0.0015$$
$$B_{00} = 0.030523$$

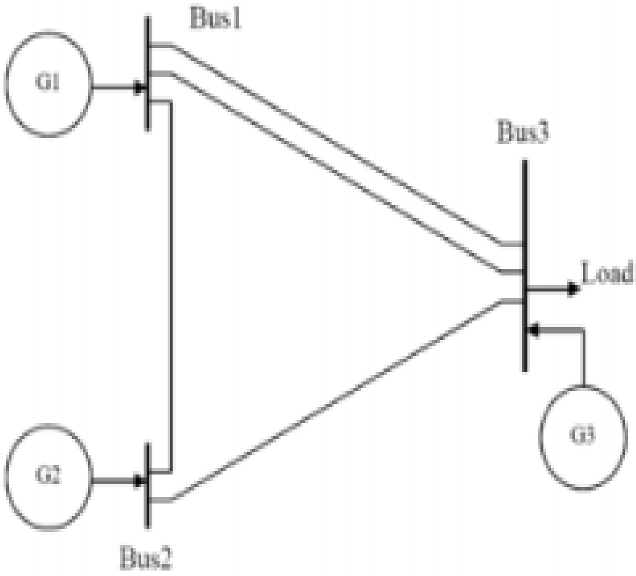

Figure 2: Typical 3-Unit system

Table 2: Data for 6-Unit System

Generating Unit	Lower Limit, P_{min} (MW)	Upper Limit, P_{max} (MW)	Cost Coefficient (a, b and c)
1	100	500	0.007, 7, 240
2	50	200	0.0095, 10, 200
3	80	300	0.009, 8.5, 300
4	50	150	0.008, 11, 200
5	50	200	0.008, 10.5, 220
6	50	120	0.0075, 12, 120

$$B_{ij} = \begin{matrix} 0.000014 & 0.000017 & 0.0000150.000019 & 0.000026 & 0.000022 \\ 0.000017 & 0.00006 & 0.0000130.000016 & 0.000015 & 0.00002 \\ 0.000015 & 0.000013 & 0.0000650.000017 & 0.000024 & 0.000019 \end{matrix}$$

$$\begin{matrix} 0.000019 & 0.000016 & 0.0000170.000071 & 0.00003 & 0.000025 \\ 0.000026 & 0.000015 & 0.000024\ 0.00003 & 0.000069 & 0.000032 \\ 0.000022 & 0.00002 & 0.0000190.000025 & 0.000032 & 0.000085 \end{matrix}$$

$$B_{0i} = 0$$
$$B_{00} = 0$$

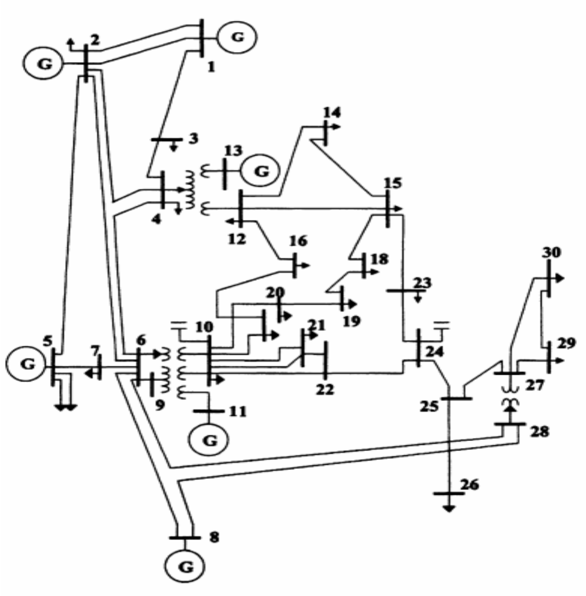

Figure 3: Typical 6-Unit system

4. RESULTS

Test Strategy

To verify the feasibility and effectiveness Genetic Algorithm (GA) in solving economic dispatch, the heuristic algorithms was applied to

1. 3-unit system with transmission losses
2. 3-unit system without transmission losses
3. 6-unit system with transmission losses
4. 6-unit system without transmission losses
5. 3-unit system with multiple fuel options

The economic dispatch problem was solved using the genetic algorithm and the performance of each generator has been judged using MATLAB 8.1.0 on an Intel(R) Pentium(R) N3540 processor, 2.16GHz with 4GB RAM.

Convergence Characteristics of Genetic Algorithm

The genetic algorithm for 3-unit system and 6-unit system are seen in figure 4 and 5.

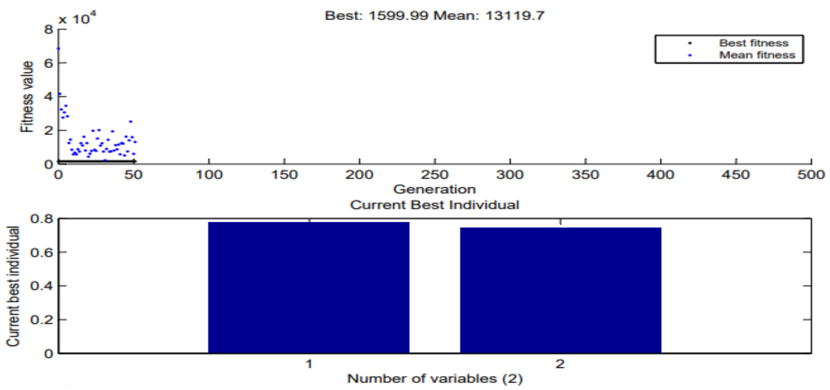

Figure 4: Output of GA for 3-unit system

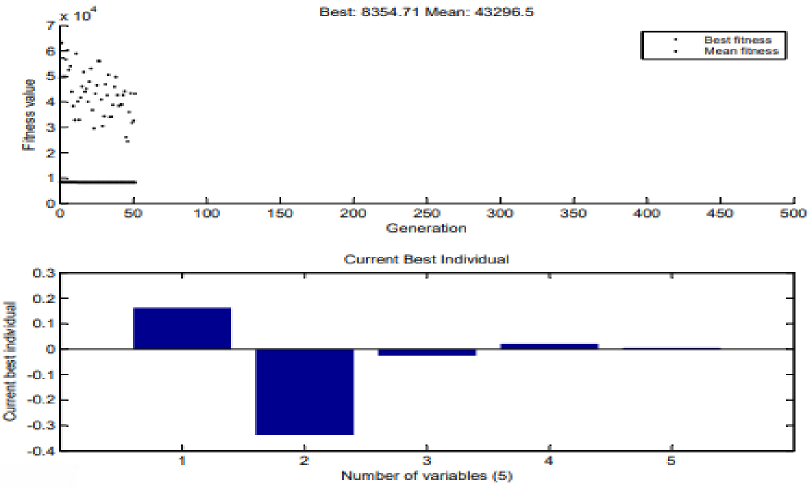

Figure 5: Output of GA for 6-unit system

Solution Quality

Genetic algorithm was tested to know its effectiveness in minimizing generation cost and as well meeting the various load demands. The first test system has three-generating units and the parameters of the 3-Unit system has transmission loss coefficient represented in B-matrix form as shown in table 1. The output of the genetic algorithm is presented in table 3. At various load demand (with 10MW increment), the result is given by genetic algorithm in term of generation cost and power loss minimization.

Table 3: Results by Genetic algorithm with transmission loss at different load values

Power Demanded (MW)	Gen 1 (MW)	Gen 2 (MW)	Gen 3 (MW)	Total Cost (#/hr.)	Power Loss (MW)
100	17.40	50.23	33.71	1220	1.34
110	20.30	51.97	39.28	1295	1.55
120	23.17	55.06	43.56	1370	1.80
130	27.54	58.30	46.24	1446	2.07
140	30.14	61.29	50.93	1523	2.36
150	31.92	65.03	55.73	1600	2.68
160	36.81	66.72	59.47	1678	3.00
170	39.85	69.82	63.68	1757	3.35
180	44.08	71.94	67.69	1836	3.71
190	47.26	76.89	70.00	1917	4.14
200	54.60	80.00	70.00	2000	4.60
210	65.08	80.00	70.00	2081	5.08
220	75.60	80.00	70.00	2167	5.60

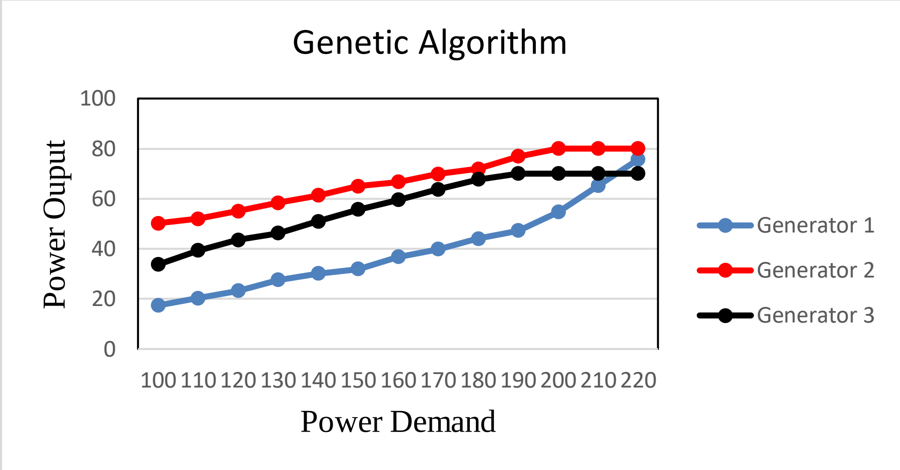

Figure 6: Power Output of three generators using Genetic Algorithm

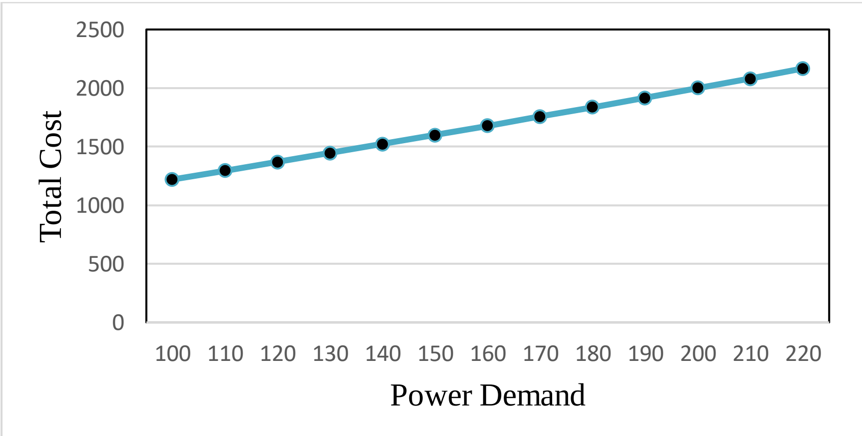

Figure 7: Total Cost using Genetic Algorithm

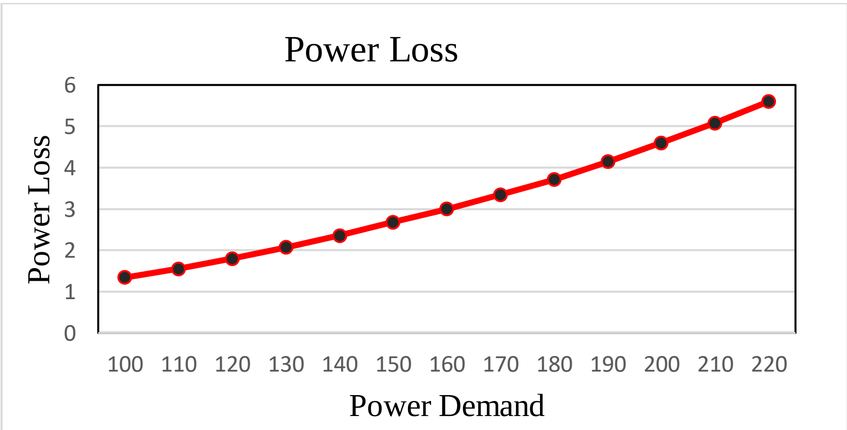

Figure 8: Total Power loss for three generators

Table 4: Result by Genetic algorithm without transmission loss at different load values

Power Demanded (MW)	Gen 1 (MW)	Gen 2 (MW)	Gen 3 (MW)	Total Cost (#/hr.)
100	15.29	53.01	31.67	1211
110	18.94	55.31	35.75	1283
120	22.19	58.38	39.43	1357
130	25.26	61.44	43.30	1431
140	28.61	64.58	46.81	1505
150	31.65	67.31	51.03	1580
160	35.53	69.60	54.87	1655
170	38.56	73.15	58.29	1731
180	42.5	74.58	62.90	1807
190	44.38	80	65.62	1884
200	50	80	70	1962
210	60	80	70	2041
220	70	80	70	2121

Increasing the number of generating units to six and using genetic algorithm to solve the economic dispatch problem of the 6-unit system, Table 2 shows the parameters of the 6-Unit system with their cost coefficient. The solution is shown in table 5.

Table 5: Result by genetic algorithm with transmission loss at different load values

Power Demanded (MW)	Unit 1 (MW)	Unit 2 (MW)	Unit 3 (MW)	Unit 4 (MW)	Unit 5 (MW)	Unit 6 (MW)	Total Cost (#/hr.)	Power Loss (MW)
500	208.84	56.97	87	51.07	50.82	50.95	6146	5.66
600	276.68	51.80	119.57	54.10	52.48	53.34	7217	7.95
700	318.04	76.05	135.37	52.25	56.64	50.83	8355	10.82
800	356.63	102.53	145.47	55.08	73.56	50.99	9561	14.25
900	388.97	113.16	199.46	67.15	98.23	51.48	10814	18.44
1000	409.26	134.94	227.19	88.11	110	53.70	12113	23.22
1100	433.89	153.07	241.90	103.94	138.11	57.76	13452	28.67
1200	456.77	172.67	256.81	115.82	158.32	74.31	14835	34.71
1300	484.98	185.76	273.59	133.83	175.53	87.57	16257	41.27
1400	499.09	200	287.10	150	200	112.79	17721	48.98

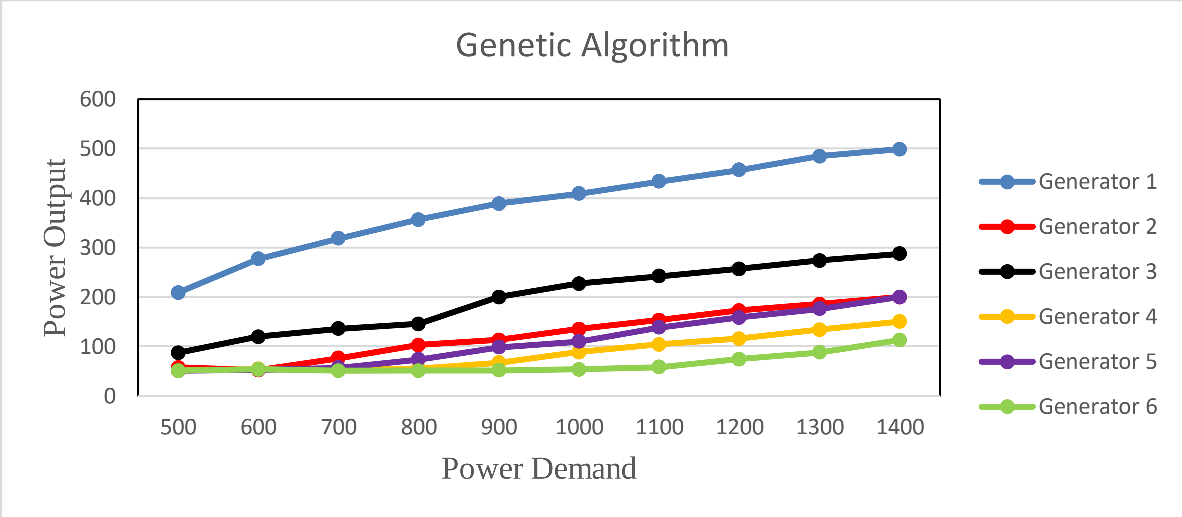

Figure 9: Power output of six generators using Genetic Algorithm

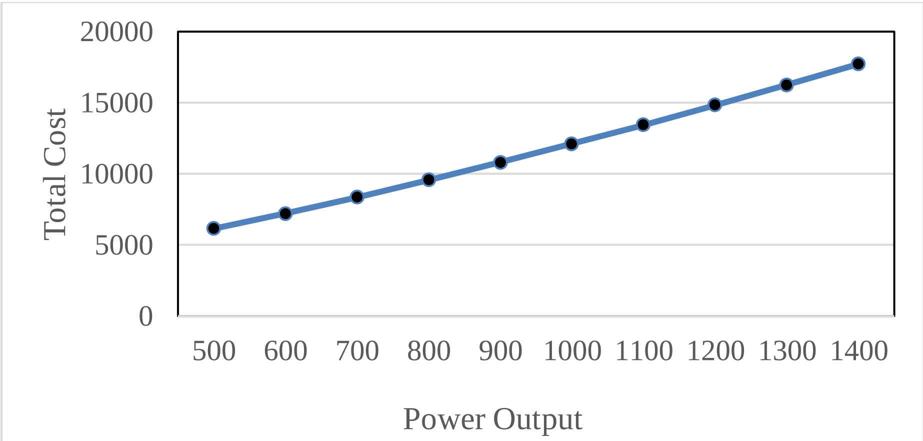

Figure 10: Total Cost of generation for six generators using Genetic Algorithm

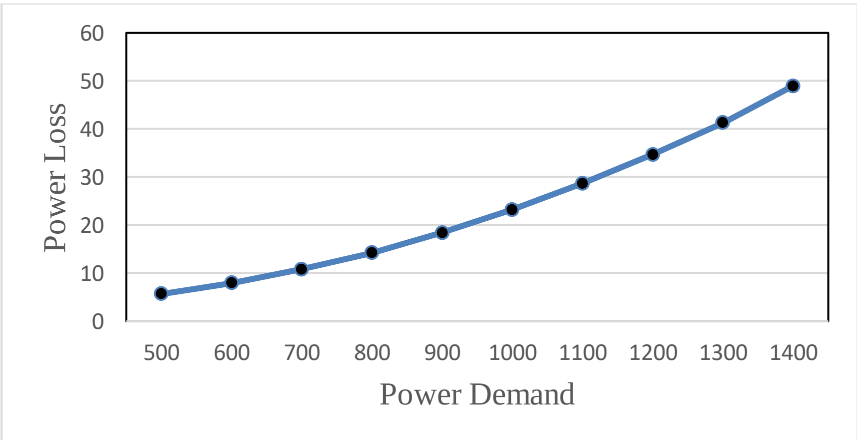

Figure 11: Total Power loss for six generators using genetic algorithm

Table 6: Result by Genetic Algorithm without transmission loss at different load values

Power Demanded (MW)	Unit 1 (MW)	Unit 2 (MW)	Unit 3 (MW)	Unit 4 (MW)	Unit 5 (MW)	Unit 6 (MW)	Total Cost (#/hr.)
500	208.63	50.89	81.05	50.86	53.75	54.81	6090
600	276.47	51.03	119.93	52.00	50.32	50.24	7121
700	310.02	71.12	158	53.83	55.40	51.63	8232
800	351.95	84.34	183.77	52.91	75.76	51.27	9391
900	368.45	119.08	200.87	63.42	95.37	52.80	10588
1000	394.39	130.48	213.53	83.83	125.77	52	11818
1100	407.83	148.14	244.64	100.31	142.83	56.26	13083
1200	434.16	163.10	252.41	114.74	164.93	170.65	14377
1300	451.77	171.43	272.30	132.26	182.12	90.13	15699
1400	470.45	193.16	282.70	148.03	200	105.66	17048

An economic dispatch problem with multiple fuel options has also been considered. Particle swarm optimization was applied to solve the problem. The parameters of the generating unit as shown in table 7. The results of the three approaches with multiple fuel option combination are shown in table 8a-8d.

Table 7: Data of 3-unit generator with multiple fuel options

Generating Unit	Lower limit, P_{min}	Upper Limit, P_{max}	FT	a	B	c
1	190	490	1	0.001066	0.8773	13.92
			2	0.001597	-0.5206	99.76
2	85	265	1	0.002758	-0.6348	52.85
			2	0.001049	0.03114	1.983
3	200	500	1	0.0002454	0.3559	43.35
			2	0.001165	-0.2267	43.77

Table 7a Genetic Algorithm (Demand =700MW)

Unit	FT	Genetic algorithm
1	1	190.28
2	1	209.02
3	1	300.69
Total Power		699.99
Total Cost		432.67

Table 7b

Unit	FT	Genetic algorithm
1	1	190.64
2	1	196.68
3	2	312.68
Total Power		700
Total Cost		341.39

Table 7c

Unit	FT	Genetic algorithm
1	1	190.18
2	2	89.04
3	1	420.78
Total Power		700
Total Cost		493.90

Table 7d

Unit	FT	Genetic algorithm
1	1	191.40
2	2	151.16
3	2	357.44
Total Power		700
Total Cost		405.49

Computational Efficiency

Some certain columns in Tables 3–6 and all tables in 7a-7d present the cost achieved by Genetic Algorithm for the three test cases with constraint satisfaction. The costs achieved by genetic algorithm however, is computationally efficient as time requirement is quite small for all cases.

5. CONCLUSION

The complex problem of economic power dispatch is solved using genetic algorithm. The test results clearly demonstrated that genetic algorithm which is capable of achieving global solutions is simple, good computational efficiency and has quite a stable dynamic convergence characteristic. In the case economic dispatch problem with multiple fuel option, the genetic algorithm has shown capability of achieving result as the conventional numerical method.

6. REFERENCES

1. A. J. Wood and B. F. Wollenberg, Power Generation, Operation and Control. New York: Wiley, 1984
2. D. C. Walter and G. B. Sheble, “Genetic algorithm solution of economicload dispatch with valve point loading,” IEEE Trans. Power Syst., vol. 8, no. 3, pp. 1325–1332, Aug. 1993.
3. N. Sinha, R. Chakrabarti, and P. K. Chattopadhyay, “Evolutionary programming techniques for economic load dispatch,” IEEE Trans. Evol.Comput., vol. 7, no. 1, pp. 83–94, Feb. 2003.
4. Wang and S. M. Shahidehpour, “Effects of ramp rate limits on unit commitment and economic dispatch,” IEEE Trans. Power Syst., vol. 8, no. 3, pp. 1341–1350, Aug. 1993.
5. https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/hmw/article1.html
6. R. Hassan, B. Kohanim, O. De weck, “A copmarison of particle swarm optimization and the genetic algorithm”
7. EMS-LECTURE 7: Economic Dispatch and Optimal Power Flow
8. S. A. Cook. ”An overview of computational complexity”, in Communication of the ACM, vol. 26, no. 6, June 1983, pp. 401–408.
9. K.S. Srikanth, D. Ravi Kishore, T. Vijay Muni, K. Naresh, and M.C. Rao “Economic Load Dispatch with Multiple Fuel Options Using GA Toolbox in Matlab” Journal of Science and Technology, Volume 1, Issue 1, PP 17-24, December 2016.

10. June Ho Park, Sung Oh Yang, Hwa Seok Lee, Young Moon Park, “Economic load dispatch using evolutionary algorithms”, IEEE Transactions on power systems, Vol. 2, pp. 441-445, 1996.
11. A. Kaur, H. Pal Singh and A. Bhardwaj “Analysis of Economic Load Dispatch Using Genetic Algorithm” International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 3, Issue 3, March 2014.
12. Jong-Bae Park, Member, Ki-Song Lee, Joong-Rin Shin, and Kwang Y. Lee, “A Particle Swarm Optimization for Economic Dispatch with Non-smooth Cost Functions” IEEE Trans. Magn. IEEE Transaction on Power Systems, Vol. 20, No. 1, February 2005.
13. A. Radhakrishnan, S. Srikrishna “Economic Dispatch with Multiple Fuel Options Using CCF”, Journal on Energy and Power Engineering, 2011, vol 3, 113-119.
14. K. S. Swaroop and P. R. Kumar, “A new evolutionary computation technique for economic dispatch with security constraints,” Int. J. Elect. Power Energy Syst., vol. 28, pp. 273–283, 2006.
15. Olulope P. K.. Optimising Power Loss on Nigeria Weak Transmission Line Using Facts Devices and Genetic Algorithm. IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), 13(3), 07-16, 2018,New York

