# Constrained Elitist Genetic Algorithm for Economic Load Dispatch: Case Study on Nigerian Power System

Sunny Orike
Department of Computer Science
Heriot-Watt University
Edinburgh, United Kingdom

David W. Corne
Department of Computer Science
Heriot-Watt University
Edinburgh, United Kingdom

## ABSTRACT

The Economic Load Dispatch problem concerns the determination of the optimal combination of power generators' outputs with the lowest generation cost for a defined level of load. The problem arises from the fact that there are rated limits of generators' power output, fuel costs of power plants are different, energy should be balanced, and plants are not located at the same distance from load centers. This work proposes the use of a novel technique - constrained elitist genetic algorithm (CEGA) in optimizing real power scheduling for the Nigerian power system. The approach was tested and evaluated against related approaches with same test data, where it exhibited superior performance to attempts so far previously reported in the literature.

## General Terms

Artificial Intelligence, Economic Load Dispatch, Electricity Generation, Genetic Algorithm, Performance, Simulation.

## Keywords

Economic Load Dispatch, Elitism, Genetic Algorithm, Optimization, Power System

## 1. INTRODUCTION

Energy is an indispensable tool for the socio-economic growth and development of any nation, and available in two forms: renewable and non-renewable [1]. The former is obtained from natural resources, and are abundant in supply. Examples include: wind, hydro, biomass, geothermal, etc; while the latter is obtained from fossil fuels. Examples are: coal, petroleum, natural gas, and nuclear power. The world's energy demand is presently on the increase and to meet up with this trend, appropriate and timely up to date studies need to be constantly carried out in the subject of power systems.

The Economic Load Dispatch (ELD) optimization problem is the determination of the optimal combination of generators' outputs with the lowest generation cost and while meeting consumers' load demand. Put in another way, it is an optimisation problem, concerned with how electrical power generating stations can meet their customers' demands while minimizing under generation/over generation, and also minimizing the generation costs (running the generating units), and meeting up several constraints.

Genetic Algorithms (GA), also referred to by the more general term Evolutionary Algorithms (EA), are a rapidly growing and active area of research in Computational Intelligence. A GA is a heuristic search algorithm based on ideas from natural evolution theory [2]. GAs have emerged as robust and successful tools in power system optimisation in recent years. The ability of GAs to search vast problem spaces and find near-optimal results within short time periods is key to the growing interest in them. It is thought that GAs may be able to generate near-optimal combinations of generators' output with lower generation cost, while meeting consumers' load demand in the electrical power system of Nigeria.

The organization of the paper is as follows: Section 2 describes the background study of the work; Section 3 summarises previous approaches that have been applied in solving ELD problems; Section 4 describes the methodology used in this work including problem formulation, details of GAs and the proposed optimisation approach – CEGA; Section 5 gives the implementation details, showing the experimental design, results and discussions; while Section 6 concludes the findings of the work.

## 2. BACKGROUND STUDY

The electrical power sector is the largest energy sector in the world as the operations of other industries and sectors depend on electricity, and both developed as well as developing countries in the world aim to establish and maintain an effective electricity sector [3]. To this end, Electrical Power Engineers are constantly engaged in all processes involved in power generation, transmission, distribution and consumption; and they encounter challenging problems in their quest to meet the demand of consumers in an economical manner [4,5]. Analysis and computation of electrical power system load flow is a systematic procedure that involves both the electricity grid, as well as other network devices such as generators, transformers, etc; and performance evaluation mechanism of the network. A typical power system network is represented in Figure 1.
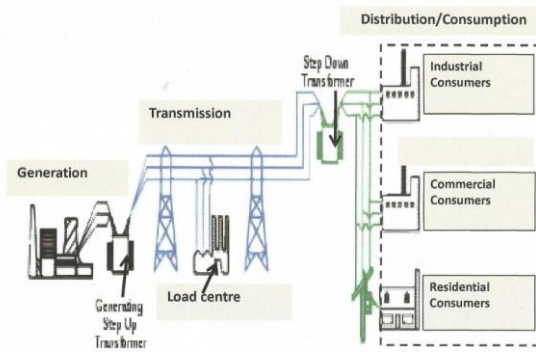
**Figure 1: A Typical Power System Representation**

No two electrical power systems are exactly the same, but they share a range of common characteristic features. As shown in Figure 1, power generation is achieved by means of generators (synchronous machines driven by turbines). The generated power is conveyed from the power stations over long distances by means of transmission network through load centres to distribution network, and finally the consumers. Electrical power is generated, transmitted and distributed by three-phase alternating current systems, with voltage and frequency levels required to remain within tolerance levels to ensure high quality output.

Interests in electrical power studies date back to the works of William Gilbert, who is regarded as the father of electricity and magnetism and also the first electrical engineer; George Ohm, who quantified the relationship between the electric current and voltage; Michael Faraday, who discovered the principle of electromagnetic induction that explains the operation of generators and transformers; and James Clerk Maxwell, who published a unified theory of electricity and magnetism [6]. The introduction of computers facilitated the approached adopted in the analysis and studies for optimum performance.

In Nigeria, the power sector has been faced with lots of problems for many years which include: low generation, poor transmission and distribution and decaying facilities [7]. The demand for electrical energy is continuously increasing from residential, commercial as well as industrial consumers, with the highest increase being in residential consumption. The efficient and optimum operation of electric power systems has always been of great concern. It is necessary for utility companies to run their power systems with minimum cost while making profits, and satisfying customers' demands at all times. For many years, the electrical power sector was managed by the Electricity Corporation of Nigeria (ECN). It was later transformed into the National Electric Power Authority (NEPA), and consequently into Power Holding Company of Nigeria (PHCN) that got liquidated in 2012. The system grid made use of in this work consists of a 31-bus, 330kV network which connects four thermal plants and three hydro plants to several load centres [8]. The major problem of the power sector is low generation [9]. With an installed capacity of 8500MW, total power generated in 2008 was 2400MW; 3700MW in 2009, and 4100MW in 2010, for a country with a population of over 160 million [10]. Major reform milestones achieved by the sector according to the Director-General, Bureau of Public Enterprises include: Creation of the Nigerian Electricity Regulatory Council as an independent regulator; creation of PHCN taking over from the

defunct NEPA, unbundling of PHCN into eighteen (18) successor generation, transmission and marketing firms and privatisation of these companies; creation of Nigerian Electricity Liability Management Company; creation of Nigerian Bulk Electricity Trading Plc [11]. Table 1 shows the sector's reform targets for generation, transmission and distribution from 2011 to 2013. In line with the vision 20:2020, the Federal Government aims a target generation of 40,000MW by 2020 [10].

**Table 1: Nigerian Power System Reform Targets**

|  | **2011** | **2012** | **2013** |
|---|---|---|---|
| Generation (MW) | 9767 | 11879 | 14218 |
| Transmission (MW) | 7488 | 8986 | 9885 |
| Distribution (MW) | 7485 | 8061 | 9057 |

## 3. PREVIOUS APPROACHES

It is very difficult to obtain solutions for ELD problems through the conventional methods of solving simultaneous equations. Solution is only possible if the number of variables is reduced by specifying values for some of the variables so that the number of variables and equations are equal. Power system problems have been solved by means of analogue and digital methods. Analogue solutions were achieved through matrix algebra, and in Nigerian Power System, using Gauss-Seidel iterative algorithm and Newton Raphson's methods [5]. But the major problems were slow rate of convergence, complex computations, more computer time per iteration, and large computer memory required. Besides, the solution was only restricted to the computation of bus voltages, bus powers and network losses for a section of the transmission grid. It had nothing to do with power generation.

Numerous traditional techniques have been reported in literature to solve ELD problem through digital methods, including: Linear and Non-linear Programming [12--14], Lambda-iterative [12,14], Integer Programming [15], Quadratic Programming [13,14], Dynamic Programming [12,13], Interior Point Method [13,14] and Lagrangian Relaxation Method [13]. In most of these algorithms, optimality of solutions was mathematically formulated, and could be extended to several optimization tasks, including, power systems planning, unit commitment, generation scheduling, etc; with high computational efficiency, with ease of implementation. They can also handle problems with hundreds and thousands of variables and constraints with inexpensive computers. However, solutions obtained using these approaches have their inherent implementation limitations [12,13]. Besides, the solutions of large-scale power systems are not very simple, and ELD problems are associated with a lot of uncertainties because they are large, complex, and geographically widely distributed. Many of the techniques such as Lambda-iterative, gradient-based and Lagrange relaxation methods are not applicable to some classes of objective functions with non-smooth cost functions, as they fail to get optimal solutions, with a possibility of being stuck in local optima [12]. For effective implementation of lambda-iterative method, the problem formulation needs to be continuous. Other problems encountered by these methods include: long execution time, poor computation efficiency, and *curse of dimensionality* (a process whereby the dimensions of the problem becomes too large that it requires massive computational effort). Hence, solutions of ELD

problems through mathematical computation methods have minimal applications.

Despite the successes from mathematical optimization methods to solving power system problems, the enumerated challenges have pushed developments of ELD in the recent decades from the traditional iterative methods to stochastic search methods, as there are still some classes of problems whose complete solutions could not be realized in a conventional setting. Most of such problems require the use of facilities to store human knowledge, operators' judgment, and experience gained with time [16]. Advancements in Computer Science and Engineering have led to greater need for more specialized artificial intelligence approaches to overcome the above limitations. Modern optimisation techniques based on this artificial intelligence approach to solving ELD problems include: Evolutionary Algorithms [8,17], Fuzzy Logic [14], Artificial Neural Networks [14,19], Expert Systems [16,18] and Swarm Intelligence [20]. These methods have the ability of searching the global optimal solutions to power systems, but however, most of them suffer from long computation time, with huge number of problem-specific parameters [12].

Evolutionary Algorithms (EAs), which solve problems in a way inspired by Charles Darwin's principle of natural evolution [2], have the ability to produce optimal or near-optimal solutions in reasonable time, and are highly flexible and adaptable to different problems. There are many varieties (e.g. Genetic Algorithms (GA), Genetic Programming (GP), Evolutionary Programming (EP), Evolution Strategies (ES), Estimation of Distribution Algorithms (EDA), Differential Evolution (DE), and so on, each tending to be associated with particular candidate solution representations (e.g. real numbers (DE), discrete variables (GA), or function trees (GP) and using distinct types of operations. ES, for example, traditionally do not implement crossover operators (in which new candidate solutions are generated by combining parts from two distinct parent solutions) [21], while GAs and GP are almost invariably used with a crossover operator. EAs are also commonly considered as a general family of optimizers that also includes heuristic local optimization methods such as hill-climbing (HC) or simulated annealing (SA) [22]. The main difference between EAs and techniques such as HC and SA is that EAs employ a population of candidate solutions that gradually evolves over time, rather than a single candidate solution that improves over time (i.e. using a population of size 1). Finally, a recent type of EA is known as EDAs or Probabilistic Model Building Genetic Algorithms (PMBGAs); these methods build probabilistic models of solutions that replace the standard operators as ways to generate new candidate solutions.

Another computational intelligence approach that is sometimes used in the context of ELD or similar problems is Fuzzy Logic (FL), which addresses the uncertainties that exist in electrical power systems, applied in areas of load forecasting, fault diagnosis, system protection and stability. Although FL represents the operational constraints of power systems more accurately, however, the approach has a limited application to active power generation, unless used as a hybrid with other approaches [18,24]. Meanwhile, a different type of computational intelligence approach, Artificial Neural Networks (ANN) uses a simple model of the construction and behavior of the human brain to solve classification and pattern recognition problems. An ANN is a layered network of artificial neurons, working in analogy to biological neurons. The main advantages of ANN are speed, robustness, ability to learn, adapt to training data, and appropriateness for non-

linear modeling. ANN has been applied to load forecasting, voltage control, economic dispatch, fault diagnosis, static and dynamic security assessment, power system stability, etc. However, it is dimensionally too large which requires a lot of trainings, and there are always results generated even with unreasonable and invalid data [19].

Expert System (ES), first proposed in the early 1970s [16], is a knowledge-based search method that uses knowledge interface processes in solving difficult problems. Solutions obtained using ES are permanent and consistent in their results, but also easily transferrable, reproduced and documented. In the past two decades, ES has been applied in the areas of power system planning: fault diagnosis, alarm processing, power protection, restoration and voltage control [18]. However, it encounters knowledge bottleneck problem, that is, it has the inability to learn and/or adapt to new situations and environments [24].

Swarm Intelligence (SI) is a branch of naturally-inspired algorithms that focus on insect behavior in developing solutions to problems [20,24]. It originated from the study of colonies, or swarms of social organisms and consists of three main algorithms: Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and Bee Colony Optimization (BCO). ACO studies artificial systems, in relation to the behavior of real ant colonies used in solving discrete optimization problems. PSO is a population-based, global optimization approach, which is modeled on the social behavior of flock of birds. BCO imitates the behavior of swarms of honey bees in food foraging. ACO is characterized by a good feedback mechanism in recovering good solutions and distributed computation, which helps in avoiding early/premature convergence. The main draw-back is poor computational ability [24]. ACO has been widely used in power system in the determination of the shortest route for transmission network [18]. PSO and BCO are still emerging approaches in ELD optimization.

In this paper, an elitist GA with penalty imposed in the form of constraints was used to solve the problem of real power ELD, tested with the 31-bus Nigerian power system grid, and compared with other approaches on the same data.

## 4. METHODOLOGY

In order to optimise electrical power systems, three basic things are required:

1. The model of the system;

2. The objective function;

3. The optimization approach to be adopted.

## 4.1 Problem Formulation

Optimisation technique for scientific and engineering systems involves the process of solving a set of non-linear equations describing the optimal or steady-state operation of the systems. The ELD problem is formulated as minimising a scalar objective function through the optimal operation of a vector of control parameters. This is mathematically illustrated in (1), (2) and (3); and diagrammatically represented in Figure 2.

Minimize:        $C(x, u)$        (1)

Subject to:        $G(x, u) = 0$        (2)

$$H\,(x,\,u) \leq 0 \qquad (3)$$

*Where:*

*C = Cost function*

*x = vector of dependent variables (generating cost);*

*u = vector of control variables (generator outputs);*

*G(x, u) = Set of non-linear equality constraints (power balance);*

*H(x, u) = Set of inequality constraints (limits in generators outputs).*
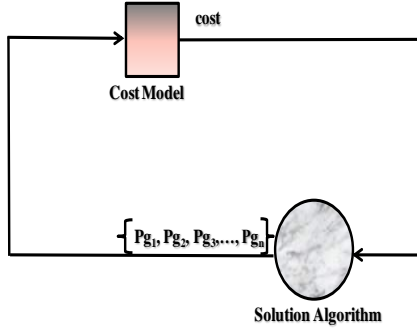


**Figure 2: A Representation of Research Methodology**

It consists of two main components: the controlling device and the device to be controlled. A typical ELD problem is one that involves the optimal set of generating units. This minimises the operation cost (mainly fuel cost). The device to be controlled is the generating cost model, while the solution algorithm (GA) is the controlling device. A vector of generators output is fed into the cost model, which produces a scalar cost of generating those outputs. The cost is passed on to the solution algorithm to be minimised, an iterative process which continues until a either a cost lower than acceptable minimum is found or number of maximum iterations is completed.

In a system of generating units connected to a transmission network, modeling and formulating the ELD problems involve three different cases: (i) Neglecting both generators' limits and power losses; (ii) Including generators' limits, but neglecting losses; (iii) Including both generators' limits and power losses. In case (i), the total power demand is equal to the sum of all generating units output, where a cost function is assumed to be known for each unit. The problem here is the computation of total generation cost, and the generator output for each unit such that the objective function is the summation of all the cost of all the generating units as shown in (4), subject to the constraint in (5).

$$C_t = \sum_{i=1}^{ng} C_i = \sum_{i=1}^{ng} (a_i + b_i P_{gi} + c_i P_{gi}^2) \qquad (4)$$

$$\sum_{i=1}^{ng} P_i = P_D \qquad (5)$$

In case (ii), an inequality constraint shown in (6) is considered because in real world, generators have minimum and maximum generating limits.

$$P_{gi}^{\min} \leq P_{gi} \leq P_{gi}^{\max} \qquad (6)$$

Case (iii) involves the power transmission losses $(P_L)$, expressed as a quadratic function of generators in (7), and the equality constraint (power balance operation) is defined in (8), in addition to the inequality constraint of (6), which is the case considered in this work.

$$P_L = \sum_{i=1}^{ng} \sum_{j=1}^{ng} (P_{gi}[B]P_{gj}) + \sum_{i=1}^{ng} P_{gi} B_o + B_{oo} \qquad (7)$$

$$\sum_{i=1}^{ng} P_{gi} = P_D + P_L \qquad (8)$$

*Where:*

| | | |
|---|---|---|
| $C_t$ | = | *Total generation cost;* |
| $C_i$ | = | *Generation cost of unit, i;* |
| *ng* | = | *Total number of power units;* |
| $a_i, b_i, c_i$ | = | *Cost coefficient of unit, i;* |
| $P_{gi}$ | = | *Power output of unit, i;* |
| $P_D$ | = | *Power demand;* |
| $P_L$ | = | *Transmission power loss;* |
| $P_{gi}^{min}$ | = | *Minimum generating limit of unit, i;* |
| $P_{gi}^{max}$ | = | *Maximum generating limit of unit, i;* |
| $B, B_o, B_{oo}$ | = | *Generalised loss formula coefficients.* |

This provides an optimal flow of real powers with the lowest operating cost. Figure 3 shows a flow diagram for the ELD optimisation.
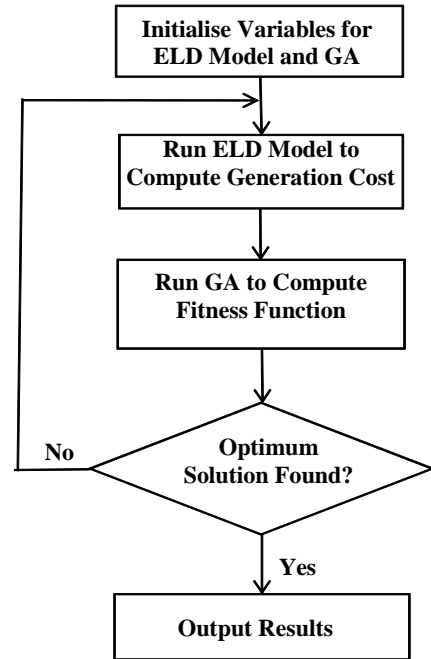


**Figure 3: Flow Diagram for the ELD Optimisation**

## 4.2 Genetic Algorithm (GA)

GA consists of computer programs which create an environment where populations of solutions, called *chromosomes* (representing possible solutions to a problem) can compete with themselves, and only the fittest among all survive through the evolutional iteration process called generations. GA combines the application of selective pressure to a diversified population [2][21].

Steps to successful GA implementation are described as follows:

a)      ***Generate*** initial random population of chromosomes consisting of double number genes within the generating units' minimum and maximum limits. These represent the generators' active power outputs;

b)      ***Evaluate*** the fitnesses of each chromosome in the population;

c)      ***Select*** parent chromosomes from the population pool;

d)      ***Breed*** the parents, by performing crossover and mutation to produce offspring (children) chromosomes;

e)      ***Replace*** the population of parents with those of the children;

f)      If the ***stopping criterion*** is reached, i.e. the best chromosome (solution) is found or the maximum generation is reached, output the best chromosome as the required solution; otherwise go back to step (b) above.

After several iterations of the algorithm, referred to as *generation*, it converges to the best solution. This resulting solution represents the optimal, solution to the problem being solved. GA is required to find the solution within a reasonable space of time. During the evolutionary process described in this work, a certain percentage of the best chromosomes in the current generation are copied onto the subsequent ones. This concept, called elitism, prevents the loss of promising solutions found in each run. This solution approach is described in the next section.

## 4.3 Optimization Approach

Elitism is a strategy that replaces the worst solutions of the new population with the best solutions of the current population. Genetic algorithm with this developed concept is called elitist genetic algorithm (EGA). The following pseudocodes were used to realise the EGA, with a work flow diagram shown in Figure 4.

*(i)*      Generate an initial population $P$ of size $N$.

*(ii)*      Evaluate the fitness of each solution, $S$.

*(iii)*      Select the best solution(s) $S_{best}$, to form the mating/breeding pool (parents).

*(iv)*      Perform breeding (crossover and mutation operations) on the parent solutions from the mating pool and obtain a new population, $P_{new}$.

*(v)*      Evaluate the fitness of each solution $S$ of $P_{new}$ with $S_{best}$. If the optimum solution is found or maximum generation is reached, output result, otherwise process elitist strategy.

*(vi)*      In the elitist process, if the fitness of each solution in $P_{new}$ is less than the fitness of $S_{best}$, replace the worst

solution of $P_{new}$ with $S_{best}$. Sort chromosomes in the resultant population. Based on elitism rate (x%) of $N$, compute the elites and copy to next generation. Go back to step (ii) above.
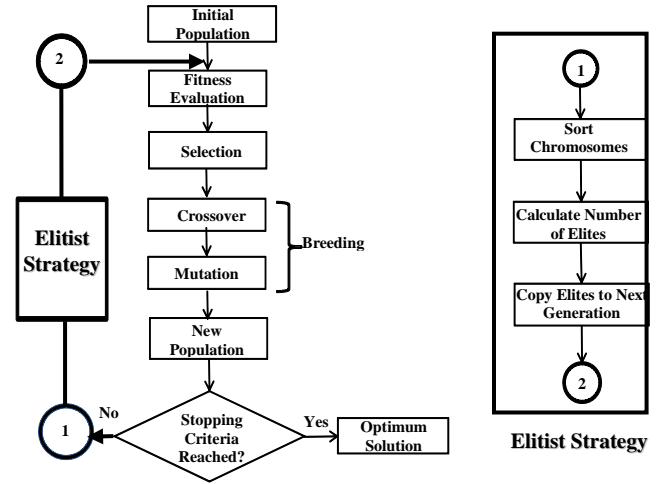


**Figure 4: Work Flow Diagram for the Elitist Genetic Algorithm**

## 5.  EXPERIMENTAL DESIGN, RESULTS AND DISCUSSION

The solution of the EGA with various constraints (equality and inequality) imposed on the ELD problem is herein referred to, in this work as *constrained elitist genetic algorithm* (CEGA). The imposed constraints are minimum/maximum generation limits and real power output balance. Implementation of the CEGA started with encoding of the problem, through the choice of appropriate parameters, and ensuring that there is sufficient power system data needed for the computation. The genetic parameters were properly tuned to make sure that only the best of them were chosen for the main experiment.

In the GA run, realisation of solution to the CEGA is by means of objective function, defined in (4), a process which calculates the fitness values of each solution. The equality constraint of (8) ensures that equilibrium is maintained between the total power generated ($P_{gi}$) and system load demand ($P_D$). In practical implementation, (8) can be re-written as an error function as in (9), with an aim to minimising it.

$$\varepsilon = \left| \sum_{i=1}^{ng} P_{gi} - P_D - P_L \right| \qquad (9)$$

Convergence was achieved when the error decreased to a tolerance level. To speed up the convergence and realise the best solution, the fitness values were normalised to a range between 0 and 1, with the fitness function defined in (10), where $k$ is a penalty factor, and $\sigma$ a scaling factor expressed as a reciprocal of the power demand, $P_D$. This process iterates until the convergence occurs. With the best solution obtained, the total power generated, total generation cost, and total transmission loss are computed.

$$f = \frac{1}{1 + (k * \varepsilon * \sigma)} \qquad (10)$$

Table 2 shows parameters for the generating capacities and cost coefficients from [8], for four of the seven thermal stations studied in the work.

**Table 2: Nigerian Thermal Stations Generating Capacities and Cost Coefficients (N = Naira)**

| *Power Station* | $P_g^{min}$ (MW) | $P_g^{max}$ (MW) | *a* (N/hr) | *B* (N/MWhr) | *c* (N/MW²hr) |
|---|---|---|---|---|---|
| Egbin | 275.0 | 1100.0 | 12787.00 | 13.10 | 0.031 |
| Sapele | 137.5 | 550.0 | 6929.00 | 7.84 | 0.130 |
| Delta | 75.0 | 300.0 | 525.74 | -6.13 | 1.200 |
| Afam | 135.0 | 540.0 | 1998.00 | 56.00 | 0.092 |

Parameters for the hydro plants (Shiroro, Kainji and Jebba) are fixed according to the respective utility's operating practices [8]. The power loss was computed using (7), with the loss formula (B-coefficients) for the Nigerian power system shown in (11), (12) and (13), as provided in [8].

$$B = \begin{bmatrix} 0.0037 & 0.0002 & -0.0074 & 0.0005 & 0.012 & -0.0076 & -0.0036 \\ 0.0002 & 0.0103 & 0.0033 & -0.0031 & 0.0022 & 0.0005 & -0.0011 \\ -0.0074 & 0.0033 & -0.0076 & 0.0023 & 0.013 & 0.0042 & 0.0152 \\ 0.0005 & -0.0031 & 0.0023 & 0.0056 & 0.0149 & 0.0004 & -0.0096 \\ 0.012 & 0.0022 & 0.013 & 0.0149 & 0.0935 & -0.0248 & -0.1354 \\ -0.0076 & 0.0005 & 0.0042 & 0.0004 & -0.0248 & 0.0127 & 0.0649 \\ -0.0036 & -0.0011 & 0.0152 & -0.0096 & -0.1354 & 0.0649 & 0.0769 \end{bmatrix} \quad (11)$$

$$B_0 = \begin{bmatrix} -0.0154 & -0.0119 & -0.0812 & -0.0881 & 0.0049 & -0.2679 & 0.2456 \end{bmatrix} \quad (12)$$

$$B_{00} = 0.2278 \quad (13)$$

The CEGA was built based on these data, using the evolutionary computation in java toolkit [25].

## 5.1 Evolution Computation in Java

The Evolutionary Computation in Java (ECJ) toolkit is a simulation package based on the Java programming language, developed by Sean Luke [25]. It consists of modules and interfaces that implement the objects and methods, including: Problem, Population, Initializer, Evaluator, Exchanger, Breeder, Fitness, Statistics, etc. The sequence of activities of the operators and modules is as follows:

(i)     Create parameters file, and define the parameters;

(ii)    Define the breeding procedure;

(iii)   Define the problem;

(iv)    Define a mechanism to run the problem by calling the parameter class.

The fitness function is problem-dependant, defined in the Problem method, with the Evaluate and Describe methods written for a particular class of problem. Table 3 shows the result of parameters tuning used in the main experiment.

## 5.2 Simulation Results

The simulation results produced are given in Table 4, which shows the optimal combination of the generators' outputs for the lowest generation cost, and power loss, while meeting the system power demand, minimizing under generation/over generation, and satisfying various equality and inequality constraints.

**Table 3: Experimental Parameters**

| Parameters | Values |
|---|---|
| Generation | 350 |
| Population size | 100 |
| Selection method | Tournament |
| Crossover type/rate | 1-point, 0.6 |
| Mutation type/rate | Single, 0.01 |
| Elitism rate | 10% |

**Table 4: Simulation Results for three GA Approaches**

| Power Stations Outputs (MW) | GA Approaches | | |
|---|---|---|---|
| | CGA | μGA | CEGA |
| Egbin ($P_{g1}$) | 877.15 | 1011.39 | 984.54 |
| Sapele ($P_{g2}$) | 194.64 | 173.28 | 178.22 |
| Delta ($P_{g3}$) | 106.49 | 111.18 | 77.72 |
| Afam ($P_{g4}$) | 371.13 | 261.43 | 303.36 |
| Shiroro ($P_{g5}$) | 490.00 | 490.00 | 490.00 |
| Kainji ($P_{g6}$) | 350.00 | 350.00 | 350.00 |
| Jebba ($P_{g7}$) | 450.00 | 450.00 | 450.00 |
| Total power | 2839.41 | 2847.28 | 2834.84 |
| System power demand | 2823.10 | 2823.10 | 2823.10 |
| **Power loss** | 16.31 | 24.18 | **11.74** |
| **Total cost (Naira/hr)** | 110,640 | 107,540 | **102,940** |

## 5.3 Evaluation

The Constrained Elitist Genetic Algorithm (CEGA) optimized more successfully than the two other Genetic Algorithm approaches reported in [8] using the same set of data – the conventional Genetic Algorithm (CGA) and the micro Genetic Algorithm (μGA). The CEGA had both the lowest generation cost of the three techniques, as well as lower power loss than either μGA or CGA. With CEGA, there are savings of N4,600 per hour over μGA and N7,700 per hour over CGA. Also, there are reductions of 4.57MW and 12.44MW in power loss over CGA and μGA respectively. The major distinguishing differences between these three GA approaches are the preservation of the "elite members" in the CEGA and copying them across generations to avoid losing promising solutions in a given generation, while μGA works on lower population size, with mutation set to 0. This leads to a quicker convergence, and may not be a good approach for higher population, where solutions might be stuck in local

optima. Also in CGA, the number of function evaluations is very large, which results to a long computation time.

# 6. CONCLUSION

In this paper, a constrained elitist genetic algorithm (CEGA) was built to solve the economic load dispatch problem of the Nigerian power system, to reduce both the transmission power loss and total cost of generation, while maintaining an acceptable generation output. Simulation results show that CEGA performed better than the micro genetic algorithm (μGA) and a conventional genetic algorithm (CGA), identified in literature with the same set of data. CEGA kept the good features of a genetic algorithm over μGA in being able to optimise problems with higher population size, while eliminating the major draw-back of CGA which is long execution time. The power dispatch is for thermal plants only; as the power allocations of hydro units were fixed in-line with the utility's operating practices. This optimization can therefore be extended to multiple fuels dispatch to give a complete solution of ELD with increased functional evaluations.

# 7. ACKNOWLEDGEMENT

# 8. REFERENCES

[1] Jager-Waldau, A. and Ossenbrink, H. 2004. Progress of Electricity from Biomass, Wind and Photovoltaics in the European Union. Renewable and Sustainable Energy Reviews. 8 (2), 157-182.

[2] Goldberg, D. E. 1989. Genetic Algorithms in Search, Optimisation and Machine Learning. Addison Wesley, Reading.

[3] Arrillaga, J. and Arnold, C. P. 1994. Computer Analysis of Power Systems. John A. Wiley and Sons Inc.

[4] Glover, J. D. and Sarma, M. S. 2002. Power System Analysis and Design. 3rd edition, Thomson Learning.

[5] Momoh, J. A. 2001. Electrical Power System: Applications of Optimizations. Marcel Dekker Inc.

[6] Stewart, J. 2001. Intermediate Electromagnetic Theory. World Scientific, 50, ISBN: 9-8102-4471-1.

[7] Chuka, E.C., Nwuba, U. and Ogonna, M.C. 2011. Optimum Reliability and Cost of Power Distribution System: A Case of Power Holding Company of Nigeria. International Journal of Engineering Science and Technology. 3 (8), 6671-6683.

[8] Bakare, G.A., Aliyu, U. O., Venayagamoorthy, G. K. and Shu'aibu, Y. K. 2005. Genetic Algorithm Based Economic Dispatch with Application to Coordination of Nigerian Thermal Power Plants. DOI: 10.1109/PES.2005.1489725.

[9] Ahiakwo, C. O. and Orike, S. 2010. Distributed Generation (Renewable Energy) - Best Option for Oil Bearing Communities. Journal of Sciences and Multidisciplinary Research. 2, 9-14.

[10] Vetiva, 2010. Nigeria: Power Sector Reform Roadmap. Vetiva Flash Note (Aug. 2010), 1-6.

[11] Bolanle, O. 2011. Nigerian Power Sector Reforms and Privatisation. Bureau of Public Enterprises, Abuja, Nigeria.

[12] Xia, X. and Elaiw, A. M. 2010, Optimal Dynamic Economic Dispatch of Generation: A Review. Journal of Electrical Power Systems Research. 80 (2010), 975-986.

[13] Kumar, C. and Alwarsamy, T. 2011. Dynamic Economic Dispatch – A Review of Solution Methodologies. European Journal of Scientific Research. 64 (4), 517-537.

[14] Pandya, K. S. and Joshi, S. K. 2008. A Survey of Optimal Power Flow Methods. Journal of Theoretical and Applied Information Technology, 450 – 458.

[15] Abdulaziz, A. U. M. and Alhabib, H. I. 2010. Power Network Planning using Mixed-Integer Programming. Journal of King Abdulaziz University Engineering Sciences. 21 (2), 15 – 34.

[16] Bansal, R. C. 2003. Literature Survey on Expert System Applications to Power Systems. International Journal Engineering Intelligent Systems. 11 (3), 103 – 112.

[17] Sayah, S. and Zehar, K. 2008. Using Evolutionary Computation to Solve the Economic Load Dispatch Problem. Leonardo Journal of Sciences. 12, 67-78.

[18] Bansal, R. C. 2005. Optimisation Methods for Electrical Power Systems: An Overview. International Journal for Emerging Electric Power Systems. 2 (1), 1 – 23.

[19] Haque, R. and Chowdhury N. 2005. An Artificial Neural Network Based Transmission Loss Allocation for Bilateral Contracts. In: Proceedings of the 18th Annual Canadian Conference on Electrical and Computer Engineering, May 1-4, 2005, 2197-2201.

[20] Kennedy, J. and Eberhart, R. C. 2001. Swarm Intelligence. Morgan Kaufmann.

[21] Mitchell, M. 1998. An Introduction to Genetic Algorithms. MIT Press.

[22] Panigrahi, C. K, Chattopadhyay, P. K., Chakrabarti, R. N. and Basu, M. 2006. Simulated annealing Technique for Dynamic Economic Dispatch. Journal of Electrical Power Components Systems Research. 34 (5), 577 – 586.

[23] Shakya, S. K. 2003. Probabilistic Model Building Genetic Algorithm: A Survey. Technical Report. Robert Gordon University, Aberdeen.

[24] Saxena, D., Singh, S. N. and Verma, K. S. 2010. Application of Computational Intelligence in Emerging Power Systems. International Journal of Engineering, Science and Technology. 2 (3), 1 – 7.

[25] Luke, S. 2008. A Java-based Evolutionary Computation Research System. Version 20. ECLab, George Mason University.