1. INTRODUCTION

Economic load dispatch (ELD) problem is one of the most important optimization problems in power systems. Main goal of economic load dispatch problem is allocation of power generation to different thermal units to minimize total fuel cost while satisfying the load demand and operating constraints. Traditionally in ELD problems, the cost function for generating units has been approximated as a quadratic function [1].

Several techniques have been introduced to solve the optimization of ELD, which can be divided into conventional and stochastic methods. Conventional methods use a deterministic approach, such as the Lagrange multiplier, Linear Programming (LP) and Dynamic Programming (DP) [2]. These methods have limitations or drawbacks when coping with more complex problems. The DP method has a problem when the number of generators is increased and higher accuracy is needed [3].

Recent techniques have been developed using stochastic approaches for solving optimization problems. Examples are an Adaptive Hopfield Neural Network [4], the Simulated Annealing method [5], and Genetic Algorithms (GA), amongst others. These new methods offer alternative techniques which attempt to overcome the drawbacks of conventional methods.

2. ECONOMIC LOAD DISPATCH

2.1 Economic load dispatch

The Economic Dispatch can be defined as the process of allocating generation levels to the generating units, so that the system load is supplied entirely and most economically. For an interconnected system, it is necessary to minimize the expenses. The economic load dispatch is used to define the production level of each plant, so that the total cost of generation and transmission is minimum for a prescribed schedule of load. The objective of economic load dispatch is to minimize the overall cost of generation.

2.2 Generator Operating Cost

The total cost of operation includes the fuel cost, cost of labor, supplies and maintenance. Generally, costs of labor, supplies and maintenance are fixed percentages of incoming fuel costs. The power output of fossil plants is increased sequentially by opening a set of valves to its steam turbine at the inlet. The throttling losses are large when a valve is just opened and small when it is fully opened.

![Simple model of a fossil plant](image1)

**Figure 1 Simple model of a fossil plant**

Figure 1 shows the simple model of a fossil plant dispatching purposes. The cost is usually approximated by one or more quadratic segments. The operating cost of the plant has the form shown in Figure 2.

![Operating cost of a fossil fired generator](image2)

**Figure 2 Operating costs of a fossil fired generator**

The fuel cost curve may have a number of discontinuities. The discontinuities occur when the output power is extended by using additional boilers, steam condensers, or other equipment. They may also appear if the cost represents the operation of an entire power station, and hence cost has discontinuities on paralleling of generators. Within the continuity range the incremental fuel cost may be expressed by a number of short line segments or piece-wise linearization. The \( P_{g_{min}} \) is the minimum loading limit below which, operating the unit proves to be uneconomical (or may be technically infeasible) and \( P_{g_{max}} \) is the maximum output limit [6].
3. PROBLEM FORMULATION

3.1 Objective Function
The objective of the economic dispatch problem is to minimize the total fuel cost while satisfying the constraints.

Fuel cost function of each thermal generating unit is expressed as a quadratic function. In terms of real power output, total cost can be expressed as the following

\[ F(P_{gi}) = a_{i}P_{gi}^2 + b_{i}P_{gi} + c_{i}, \]  

Where \( a_{i}, b_{i}, c_{i} \) are cost coefficients for \( i \)th unit. \( F(P_{gi}) \) is the total cost of generation. \( P_{gi} \) is the generation of \( i \)th unit.

3.2 Constraints
3.2.1 Power balance constraints
Generated power should be the same as the total load demand \( P_{D} \) (in MW). In this case, the active power balance is

\[ P_{D} = \sum_{i=1}^{n} P_{gi} \]  

Where \( n \) is number of total generating units.

The transmission and generator losses have been neglected.

3.2.2 Generation limits
The output power of each generating unit has a lower and upper bound so that it lies in between these bounds.

\[ P_{gi \text{min}} \leq P_{gi} \leq P_{gi \text{max}} \]   .........3)

Where \( P_{gi \text{min}} \) and \( P_{gi \text{max}} \) are the minimum and maximum output of generator \( i \), respectively.

4. LAMBDA ITERATION METHOD
The solution to this problem can be approached by considering a graphical technique for solving the problem and then extending this into the area of computer algorithms. The lambda-iteration procedure converges very rapidly for this particular type of optimization problem. The actual computational procedure is slightly more complex [7]. We use following MATLAB code formulated for no losses as:

```
clear;
class;
disp ('The minimum generation limit for generator 1 is 30 MW and maximum is 175 MW');
disp ('The minimum generation limit for generator 2 is 20 MW and maximum is 125 MW');
IFC = [0.2 40; 0.4 30];
disp ('The minimum and maximum values of lambda for the individual generators are');
lam_min_Pg1 = (IFC(1,1) * 30) + IFC(1,2);
lam_max_Pg1 = (IFC(1,1) * 175) + IFC(1,2);
lam_min_Pg2 = (IFC(2,1) * 20) + IFC(2,2);
lam_max_Pg2 = (IFC(2,1) * 125) + IFC(2,2);
disp ('The minimum load demand = 30 + 20 = 50 MW');
disp ('The maximum load demand = 175 + 125 = 300 MW');
load = 50;
Pg = [0, 0];
lambda = lam_min_Pg2;
del_lambda = 0.01;
result = [0];
for (i = 1:1:26)
    lambda = lambda + del_lambda;
    if (load - Pg(1) - Pg(2)) > 0
        lambda = lambda - del_lambda;
    else
        lambda = lambda - del_lambda;
        Pg(j) = (lambda - IFC(j,2))/IFC(j,1);
    end
    result(i,2) = lambda;
    result(i,3) = Pg(1);
    result(i,4) = Pg(2);
    result(i,5) = Pg(1)+Pg(2);
    result(i,6) = load;
    disp ('Sr.no. lambdaPg(1) Pg(2) Pg(1)+Pg(2) Load');
    disp (result);
end
```

5. GENETIC ALGORITHM
5.1 Introduction
The basic principles of GA were first proposed by Holland [8]. Thereafter, a series of literature [9], [10], [11] and reports became available. A Genetic Algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are categorized as global search heuristics. Genetic algorithms are a particular class of Evolutionary Algorithms (EA) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.

5.2 Overview of Genetic Algorithm
GA is a method for deriving from one population of “chromosomes” (e.g., strings of ones and zeroes, or bits) a new population. This is achieved by employing “natural selection” together with the genetics inspired operators of recombination (crossover), mutation, and inversion. Each chromosome consists of genes (e.g. bits), and each gene is an instance of a particular allele (e.g., 0 or 1). The selection operator chooses those chromosomes in the population that will be allowed to reproduce, and on average those chromosomes that have a higher fitness factor (defined below), produce more offspring than the less fit ones. Crossover swaps subparts of two chromosomes, roughly imitating biological recombination between two single chromosome (“haploid”) organisms; mutation randomly changes the allele values of some locations (locus) in the chromosome; and inversion reverses the order of a contiguous section of chromosome.

5.3 PROPERTIES OF GA

- Generally good at finding acceptable solutions to a problem reasonably quickly
- Free of mathematical derivatives
- No gradient information is required
- Free of restrictions on the structure of the evaluation function
- Fairly simple to develop

6. RESULTS AND DISCUSSIONS
Lambda iteration method is implemented on case of without losses and with generation limits.

Incremental fuel costs in rupees per MWhr for a plant consisting of two units are

\[ dF_{i}(P_{j})/dP_{i} = 0.20P_{i} + 40 \text{ unit of cost/MWhr} \]

\[ dF_{j}(P_{j})/dP_{j} = 0.40P_{j} + 30 \text{ unit of cost/MWhr} \]
and the generator limits are

\[ 30 \text{ MW} \leq P_1 \leq 175 \text{ MW} \quad \text{and} \quad 20 \text{ MW} \leq P_2 \leq 125 \text{ MW} \]

**Output:**

The minimum generation limit for generator 1 is 30 MW and maximum is 175 MW.

The minimum generation limit for generator 2 is 20 MW and maximum is 125 MW.

The minimum and maximum values of lambda for the individual generators are

\[
\begin{align*}
\text{lam}_{\text{min}}_{\text{Pg}1} & = 46 \\
\text{lam}_{\text{max}}_{\text{Pg}1} & = 75 \\
\text{lam}_{\text{min}}_{\text{Pg}2} & = 38 \\
\text{lam}_{\text{max}}_{\text{Pg}2} & = 80
\end{align*}
\]

The minimum load demand = 30 + 20 = 50 MW

The maximum load demand = 175 + 125 = 300 MW

<table>
<thead>
<tr>
<th>Sr.no.</th>
<th>lambda</th>
<th>Pg(1)</th>
<th>Pg(2)</th>
<th>Pg(1)+Pg(2)</th>
<th>Load</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>43.2100</td>
<td>16.0500</td>
<td>33.0250</td>
<td>49.0750</td>
<td>50.0000</td>
</tr>
<tr>
<td>2.0000</td>
<td>44.5400</td>
<td>22.7000</td>
<td>36.3500</td>
<td>59.0500</td>
<td>60.0000</td>
</tr>
<tr>
<td>3.0000</td>
<td>45.8700</td>
<td>29.3500</td>
<td>39.6750</td>
<td>69.0250</td>
<td>70.0000</td>
</tr>
<tr>
<td>4.0000</td>
<td>47.2100</td>
<td>36.0500</td>
<td>43.0250</td>
<td>79.0750</td>
<td>80.0000</td>
</tr>
<tr>
<td>5.0000</td>
<td>48.5400</td>
<td>42.7000</td>
<td>46.3500</td>
<td>89.0500</td>
<td>90.0000</td>
</tr>
<tr>
<td>6.0000</td>
<td>49.8700</td>
<td>49.3500</td>
<td>49.6750</td>
<td>99.0250</td>
<td>100.0000</td>
</tr>
<tr>
<td>7.0000</td>
<td>51.2100</td>
<td>56.0500</td>
<td>53.0250</td>
<td>109.0750</td>
<td>110.0000</td>
</tr>
<tr>
<td>8.0000</td>
<td>52.5400</td>
<td>62.7000</td>
<td>56.3500</td>
<td>119.0500</td>
<td>120.0000</td>
</tr>
<tr>
<td>9.0000</td>
<td>53.8700</td>
<td>69.3500</td>
<td>59.6750</td>
<td>129.0250</td>
<td>130.0000</td>
</tr>
<tr>
<td>10.0000</td>
<td>55.2100</td>
<td>76.0500</td>
<td>63.0250</td>
<td>139.0750</td>
<td>140.0000</td>
</tr>
<tr>
<td>11.0000</td>
<td>56.5400</td>
<td>82.7000</td>
<td>66.3500</td>
<td>149.0500</td>
<td>150.0000</td>
</tr>
<tr>
<td>12.0000</td>
<td>57.8700</td>
<td>89.3500</td>
<td>69.6750</td>
<td>159.0250</td>
<td>160.0000</td>
</tr>
<tr>
<td>13.0000</td>
<td>59.2100</td>
<td>96.0500</td>
<td>73.0250</td>
<td>169.0750</td>
<td>170.0000</td>
</tr>
<tr>
<td>14.0000</td>
<td>60.5400</td>
<td>102.7000</td>
<td>76.3500</td>
<td>179.0500</td>
<td>180.0000</td>
</tr>
<tr>
<td>15.0000</td>
<td>61.8700</td>
<td>109.3500</td>
<td>79.6750</td>
<td>189.0250</td>
<td>190.0000</td>
</tr>
<tr>
<td>16.0000</td>
<td>63.2100</td>
<td>116.0500</td>
<td>83.0250</td>
<td>199.0750</td>
<td>200.0000</td>
</tr>
<tr>
<td>17.0000</td>
<td>64.5400</td>
<td>122.7000</td>
<td>86.3500</td>
<td>209.0500</td>
<td>210.0000</td>
</tr>
<tr>
<td>18.0000</td>
<td>65.8700</td>
<td>129.3500</td>
<td>89.6750</td>
<td>219.0250</td>
<td>220.0000</td>
</tr>
<tr>
<td>19.0000</td>
<td>67.2100</td>
<td>136.0500</td>
<td>93.0250</td>
<td>229.0750</td>
<td>230.0000</td>
</tr>
<tr>
<td>20.0000</td>
<td>68.5400</td>
<td>142.7000</td>
<td>96.3500</td>
<td>239.0500</td>
<td>240.0000</td>
</tr>
<tr>
<td>21.0000</td>
<td>69.8700</td>
<td>149.3500</td>
<td>99.6750</td>
<td>249.0250</td>
<td>250.0000</td>
</tr>
<tr>
<td>22.0000</td>
<td>71.2100</td>
<td>156.0500</td>
<td>103.0250</td>
<td>259.0750</td>
<td>260.0000</td>
</tr>
<tr>
<td>23.0000</td>
<td>72.5400</td>
<td>162.7000</td>
<td>106.3500</td>
<td>269.0500</td>
<td>270.0000</td>
</tr>
<tr>
<td>24.0000</td>
<td>73.8700</td>
<td>169.3500</td>
<td>109.6750</td>
<td>279.0250</td>
<td>280.0000</td>
</tr>
<tr>
<td>25.0000</td>
<td>75.2000</td>
<td>176.0000</td>
<td>113.0000</td>
<td>289.0000</td>
<td>290.0000</td>
</tr>
<tr>
<td>26.0000</td>
<td>76.5400</td>
<td>182.7000</td>
<td>116.3500</td>
<td>299.0500</td>
<td>300.0000</td>
</tr>
</tbody>
</table>

**7. CONCLUSION**

This paper gives overview of economic load dispatch problems and solution methodologies.

Implementation is done using MATLAB programming and results are given in tabular form. Conventional method like lambda iteration method converges rapidly but complexities increase as system size increases also lambda method always requires that one be able to find the power output of a generator, given an incremental cost for that generator.

**REFERENCES**