Implementation of a New Methodology for ELD Problems

Ramesh Chevuru¹, G. V. Martheswar²

¹Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India
²Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India

Abstract: Now a day’s an electrical utility wants to maximize its profit; the optimization of economic dispatch is of economic value to the network operator. So, objective is to optimize the total cost of the plant. The ELD problem in a power system to determine the least generation of all the operating generators which will minimize the total fuel cost of the plant. This paper introduce a bio inspired and novel technique optimization algorithm called BAT ALGORITHM to solve economic dispatch problems is presented and enhancing the convergence property to obtain results quickly. The proposed method has good convergence property and better in quality of solution than PSO and IWD reported in recent literature. BAT algorithm is easy to implement and priory in terms of accuracy and efficiency compared with other algorithms.

Keywords: ELD, PSO, IWD, BAT Algorithm, Convergence.

1. Introduction

Since an engineer is always concerned with the cost of products and services, the efficient optimum economic operation and planning of electric power generation system have always occupied an important position in the electric power industry. The classic problem is the economic load dispatch of generating systems to achieve minimum operating cost. For the purpose of optimum economic operation of this large scale system, modern system theory and optimization techniques are being applied with the expectation of considerable cost savings.

The economic load dispatch (ELD) is an important function in modern power system like unit Commitment, Load Forecasting, Available Transfer Capability (ATC) calculation, Security Analysis, Scheduling of fuel purchase etc. Intelligent methods are iterative techniques that can search not only local optimal solutions but also a global optimal solution depending on problem domain and execution time limit. Among these methods, some of them are genetic algorithm (GA) [7] and [8], evolutionary programming (EP) [9] and [10], dynamic programming (DP) [11], tabu search [12], hybrid EP [13], neural network (NN) [14], adaptive Hopfield neural network (AHNN) [15], particle swarm optimization (PSO) [3]and [15], etc. In this paper the BAT ALGORITHM is proposed as a methodology for economic load dispatch. It requires less computation time and memory and the results are compared with the iteration methods PSO and IWD.

2. Economic Operation of Power System

The Economic Load Dispatch (ELD) 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. In the conventional methods, it is difficult to solve the optimal economic problem if the load is changed. The fuel cost curve in the active power generation, takes up a quadratic form, given as:

\[ F(P_{gi})=a_iP_{gi}^2+b_iP_{gi}+c_i \text{ Rs/hr} \]  

(1)

Where \(a_i,b_i,c_i\) are cost coefficient for \(i^{th}\) unit.

\(F(P_{gi})\) is the total cost of generation

\(P_{gi}\) is the generation of \(i^{th}\) plant

The optimization problem can be therefore be stated as:

\[ \text{Minimize: } F(P_{gi}) = \sum_{i=1}^{NG} F_i(P_{gi}) \]  

(2)

2.1 Economic load dispatch with losses

Over long distances, the transmission losses are a major factor and affect the optimum dispatch of generation. The transmission power loss \(P_L\) for the objective function is thus formulated as:

\[ P_L = \sum_{i=1}^{NG} \sum_{j=0}^{NG} B_{ij} P_{gi} P_{gj} \text{ MW} \]  

(3)

Where

\(B_{ij}\) are the loss coefficients or B-coefficients

\(P_{gi}\) and \(P_{gj}\) are the real power generations at \(i^{th}\) and \(j^{th}\) buses respectively. The transmission loss formula of Eq. (3) is known as George’s formula. Using the Lagrange multiplier \(\lambda\), the augmented function is,

\[ L(P_{gi}, \lambda) = F(P_{gi}) + \lambda (P_{D} - \sum_{i=1}^{NG} P_{gi}) \]  

(4)

Equation (4) shows that the minimum cost is obtained when the incremental cost of each plant multiplied by its penalty factor is same for all plants.
Finally the $P_{gi}$ can be solved by following
\[
P_{gi} = \frac{-\sum_{i=1}^{NG} 2Bi P_{ij} + \lambda \sum_{i=1}^{NG} 2Bi}{2\lambda + 2Bi} \quad (5)
\]
For any particular value of $\lambda$, above equation can be solved iteratively by assuming initial values of $P_{gi}$'s. Iterations are stopped when $P_{gi}$'s converge within specified accuracy.

2.2 Economic load dispatch with valve point loading

The real input-output characteristics display higher-order nonlinearities and discontinuities due to valve-point loading in fossil fuel burning plant. The valve-point loading effect has been modelled in as a recurring rectified sinusoidal function. Mathematically, economic load dispatch problem considering valve point loading is defined as:

Minimize operating cost:
\[
F (P_{gi}) = \sum_{i=1}^{NG} (a_i P_{i1}^3 + b_i P_{i1} + c_i + |d_i * \sin (e_i * p_{i1} min - p_{i1})|) \quad (6)
\]
Where
\[
a_i, b_i, c_i, d_i, e_i \text{ are cost coefficients of the } i^{th} \text{ unit.}
\]
Subject to:
(i) The energy balance equation given by
\[
\sum_{i=1}^{NG} P_{gi} = P_D + P_L \quad (7)
\]
(ii) The inequality constraints given by
\[
P_{gi \text{ min}} \leq P_{gi} \leq P_{gi \text{ max}} \quad (8)
\]

3. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is one of the modern heuristic algorithms, which can be effectively used to solve nonlinear and non-continuous optimization problems. It is a Population-based search algorithm and searches in parallel using a group of particles similar to other AI-based optimization techniques.

3.1 Basic particle swarm optimization

The position of each particle is represented by $P$ and also its velocity is expressed by $V$. Each particle knows its best value so far (pbest) and its position. Moreover, each particle also knows the best value so far in the group (gbest) among $p_{best}$'s. The modified velocity and position of each particle can be calculated using the current velocity and the distance from best previous position of each particle ($P_{bij}$) to best particle among all the particles in the group ($G_j$). Velocity and position of each particle can be modified by the following equation:

\[
V_{ij}^{r+1} = wV_{ij}^r + C_1 P_{ibj} + C_2 G_j \quad (9)
\]
where
\[
V_{ij}^r = P_{ij}^r + V_{ij}^r
\]
(ii)

Suitable selection of inertia weight $w$ provides balance between global and local explorations, thus requiring less iteration on average to find a sufficiently optimal solution.

3.2 Algorithm for ELD using PSO

The search procedure for calculating the optimal generation quantity of each unit is summarized as follows:

1. In the ELD problems the number of online generating units is the 'dimension' of this problem. The particles are randomly generated between the maximum and the minimum operating limits of the generators and represented using Equation as follows
\[
P_{gi} = \{P_{i1}, P_{i2}, P_{i3} \ldots \ldots \ldots P_{id}\} \quad (i=1, 2, 3, \ldots \ldots n)
\]
Where, $n$ means population size, $d$ is the number of generator, $P_{id}$ is the generation power output of d th unit at i th particle. The dimension of a population is (n * d). These genes in each individual are represented as real values.

2. To each individual of the population calculate the dependent unit output from the power balance.
3. Calculate the evaluation value of each particle $P_{gi}$ in the population using the evaluation function given by equation (2).
4. Compare each particle's evaluation value with its pbest. The best evaluation value among them pbest is identified as gbest.
5. Modify the Velocity of each particle by using the Eq. (9).
6. Check the velocity constraints of the members of each particle from the following conditions:

\[
P_{gi} = \{P_{i1}, P_{i2}, P_{i3} \ldots \ldots \ldots P_{id}\} \quad (i=1, 2, 3, \ldots \ldots n)
\]
Where
\[
P_{ij}^r = P_{ij}^r + V_{ij}^{r+1}
\]
If $V_{ij}^{r+1} > V_{ij}^{\text{max}}$, then $V_{ij}^{r+1} = V_{ij}^{\text{max}}$
If $V_{ij}^{r+1} < V_{ij}^{\text{max}}$, then $V_{ij}^{r+1} = V_{ij}^{\text{max}}$
Where $V_{ij}^{\text{max}} = -0.5 \, P_{ij}^{\text{max}}$
Where $V_{ij}^{\text{max}} = 0.5 \, P_{ij}^{\text{max}}$

7. Modify the position of each particle using the Eq. (10).
8. Modify the position of each particle using the Eq. (10).
9. If the number of iterations reaches the maximum, then go to step 10. Otherwise, go to step 2.
10. The individual that generates the latest gbest is the optimal generation power of each unit with the minimum total generation cost.

4. Intelligent Water Drops

4.1. Introduction

The IWD algorithm was first introduced by Dr. Shah-Hosseini in the year 2007 [9]. The algorithm so far was successfully implemented to the Travelling Salesman problem, n-Queens puzzle, Multidimensional Knapsack problem (MKP), Smooth trajectory planning, Robot Path planning, Vehicle routing problem and Economic Load Dispatch problem [13]. It is a nature-inspired optimization algorithm inspired from the natural water drops which change their environment to find the near optimal or optimal path to their estimation, ocean or pond. The processes that happen between the water drops of a river and the soil of the river bed formed the basis for this algorithm. IWD algorithm falls in the category of Swarm-Based Optimization algorithms.

4.2. Basic intelligent water drops

We can develop an artificial water drop which possesses the same functionality as a natural water drop. This artificial water drop can be termed as Intelligent Water Drop or IWD for short. The IWDs in the IWD algorithm are created with two main properties:

1. The amount of soil it carries, Soil (IWD).
2. The velocity that it possesses, Velocity (IWD).

Both of these properties change as the IWD flows in its environment. Consider an IWD moving in discrete finite-length steps in its environment, from its current location i to its next location j, the IWD velocity, Velocity (IWD), is increased which is nonlinearly proportional to the inverse of the soil between the two locations i and j, soil(i, j), as shown in equation

$$\Delta \text{Velocity(IWD)} = \alpha \text{NL} \times \frac{1}{\text{soil}(i,j)}$$

(13)

Here, nonlinearly proportionality is denoted by $\alpha \text{NL}$: One possible formula, according to [14] is given in eq. (4.2) in which the velocity of the IWD denoted by vel(IWD) is updated by the amount of soil, soil(i, j), between the two locations i and j:

$$\Delta \text{Vel}^{IWD}(t) = \frac{av}{bv+c+\text{soil}(i,j)}$$

(14)

Again the parameters in this equation, $a_v$, $b_v$, $c_v$, and $\alpha$ are constant velocity updating parameters that are set for a given problem. The updated velocity of the IWD, vel(IWD) $(t+1)$ after reaching node j will be equivalent to vel(IWD) $(t) + \Delta \text{YHO} (t)$. The amount of soil carried by the IWD, soil(IWD), is increased by removing some soil from the path joining the two locations i and j. The amount of the soil added to this motion can be considered as a linear motion and thus the duration of time for the IWD can be calculated by simple laws of physics for linear motion. Thus, the time taken is inversely proportional to the velocity of the IWD, velocity (IWD), and also to the distance between the two positions, d(i,j).

5. Bat Algorithm

5.1 Introduction

Bat Algorithm is a metaheuristic optimization algorithm developed by Xin-She Yang in 2010. The Bat algorithm based on the echolocation behaviour of bats.

5.2 Echolocation behaviour of bats

Bats are the only mammals with wings and have the capability of echolocation. They emit a very loud sound pulse and listen for the echo that bounces back from the surrounding objects. Their pulses vary in properties and can be correlated with their hunting strategies, depending on the species. Most bats use short, frequency-modulated (FM) signals. The typical range of frequencies for most bat species are from 25KHz to 100KHz. When hunting for prey, the rate of pulse emission can be increase up to 200 pulses per second, when they fly near their prey.

Microbats use the time delay from the emission and detection of the echo. The time difference between their two ears, and the loudness variations of the echoes will build up the three dimensional scenario of the surrounding.
They can detect the distance and orientation of the target, type of the prey and even the moving speed of the prey.

5.3 Idealized rules for BAT Algorithm

- All bats use echolocation to sense distance, and they also know the food/prey and the obstacles, in some magical way.
- Bats fly randomly with velocity \( V_i \) at position \( X_i \) with a fixed frequency \( f_{\text{min}} \), varying wavelength \( \lambda \) and loudness \( A_0 \) to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission \( r \) in the range of [0, 1], depending on the distance of their target.
- Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) \( A_0 \) to a minimum constant value \( A_{\text{min}} \). In addition to these simplified assumptions, we also use the following approximations. In general, the frequency \( f \) in a range of \([f_{\text{min}}, f_{\text{max}}]\) corresponds to a range of wavelengths \([\lambda_{\text{min}}, \lambda_{\text{max}}]\). For simplicity, we can assume \( f \) is within \([0, f_{\text{max}}]\).
- We know that higher frequencies have shorter wavelengths and travel a shorter distance. The rate of pulse emission can be simplified in the range of [0, 1], where 0 means no pulse at all, and 1 means maximum rate of pulse emission.

5.4 Movement of bats

The movement of the bats depending upon the velocity changes with respect to time step. The new solutions \( x_i^{t+1} \) and velocities \( v_i^{t+1} \) at time step \( t \) are given by:

\[
\begin{align*}
  f_i^{t+1} &= f_{\text{min}} + (f_{\text{max}} - f_{\text{min}}) * \beta \\
  v_i^{t+1} &= v_i^{t} + (x_i^{t} - x*) * f_i^{t} \\
  x_i^{t+1} &= x_i^{t} + v_i^{t+1}
\end{align*}
\]  (16-18)

Where, \( \beta \in [0, 1] \) is a uniformly distributed random vector. \( x* \) is the current global best location (solution), which is located after comparing all the solutions among all the \( n \) bats. We can choose \( f_{\text{min}}, f_{\text{max}} \) depending upon the domain size of the optimization problem. Initially, each bat is randomly assigned a frequency within the range \([f_{\text{min}}, f_{\text{max}}]\).

For the local search, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using random walk:

\[
x_{\text{new}} = x_{\text{old}} + \epsilon \times A_i^{t+1}
\]  (19)

Where, \( \epsilon \in [-1, 1] \) is a random number, while \( A_i^{t+1} = <A_i^{t}> \) is the average loudness of all the bats at this time \( t \).

5.5 Loudness and pulse emission

The loudness \( A_i \) and the rate of pulse emission \( r_i \) are updated accordingly as the iterations proceed. Once a bat has found its prey loudness decreases and rate of pulse emission increases. Assume \( A_{\text{min}} = 0 \), means that a bat has just found the prey and temporarily stop emitting any sound.

\[
\begin{align*}
  A_i^{t+1} &= \alpha A_i^{t} \\
  r_i^{t+1} &= r_i^{t} [1 - \exp (-\gamma t)]
\end{align*}
\]  (20-21)

Where, \( \alpha \) and \( \gamma \) are constants.

5.6 Pseudo code for BAT Algorithm:

BEGIN
Objective function \( f(x_i, x = (x_1, x_2, x_3, \ldots, x_d)^T \)
Initialize the bat population \( x_i, (i = 1, 2, 3 \ldots n) \) and \( v_i \)
Define pulse frequency \( Q_i \) at \( x_i \)
Initialize pulse rates \( r_i \) and loudness \( A_i \)
WHILE (\( t < \) Max. number of iterations)
Generate new solutions by adjusting frequency
\( f_i = f_{\text{min}} + (f_{\text{max}} - f_{\text{min}}) * \beta \)
and updating velocity \( v_i^t = v_i^{t-1} + (x_i^t - x*) * f_i \)
and updating location \( x_i^t = x_i^{t-1} + v_i^t \)
if (\( \text{rand} > r_i \))
Select a solution among the best solutions
Generate a local solution around the selected best solutions
end if
Generate new solutions by flying randomly
if (\( \text{rand} < A_i, f(x_i) < f(x*) \))
Accept the new solutions
Increase \( r_i \) and decrease \( A_i \)
end if
Rank the bats and find the current best \( x* \)
end while
Post process results and visualization
END

6. Case Study and Results

6.1 Case study on 6 thermal unit systems

The different methods discussed earlier are applied to one six thermal unit system to find out the minimum cost for any demand. Results of BAT Algorithm are compared with the conventional methods Particle Swarm Optimization (PSO) and Intelligent Water Drop (IWD). In this case the only transmission losses and prohibited zone constraints are considered. The maximum number of iterations has been taken as 500. All these simulation are done on MATLAB 2011 environment.

![Figure 1: Single line diagram of 6-unit thermal plant](image-url)
Table 1: capacity and coefficients data of six unit thermal system

<table>
<thead>
<tr>
<th>unit</th>
<th>P_{max} (MW)</th>
<th>P_{min} (MW)</th>
<th>a_i (RS/MW^2h)</th>
<th>b_i (RS/MWh)</th>
<th>c_i (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>500</td>
<td>100</td>
<td>0.0070</td>
<td>7</td>
<td>240</td>
</tr>
<tr>
<td>2</td>
<td>150</td>
<td>50</td>
<td>0.0090</td>
<td>11</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>200</td>
<td>50</td>
<td>0.0080</td>
<td>10.5</td>
<td>220</td>
</tr>
<tr>
<td>4</td>
<td>120</td>
<td>50</td>
<td>0.0075</td>
<td>12</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>300</td>
<td>80</td>
<td>0.0090</td>
<td>8.5</td>
<td>220</td>
</tr>
<tr>
<td>6</td>
<td>200</td>
<td>50</td>
<td>0.0095</td>
<td>10</td>
<td>200</td>
</tr>
</tbody>
</table>

The Six generating units considered are having different characteristic. Their cost function characteristics are given by following equations:

\[
\begin{align*}
F_1 &= 0.0070 P_1^2 + 7 P_1 + 240 \text{ RS/Hr} \\
F_2 &= 0.0090 P_2^2 + 11 P_2 + 200 \text{ RS/Hr} \\
F_3 &= 0.0080 P_3^2 + 10.5 P_3 + 220 \text{ RS/Hr} \\
F_4 &= 0.0075 P_4^2 + 12 P_4 + 120 \text{ RS/Hr} \\
F_5 &= 0.0090 P_5^2 + 8.5 P_5 + 220 \text{ RS/Hr} \\
F_6 &= 0.0095 P_6^2 + 10 P_6 + 200 \text{ RS/Hr}
\end{align*}
\]

Table 2: Prohibited zone limits of six unit system

<table>
<thead>
<tr>
<th>Unit</th>
<th>Zone-1</th>
<th>Zone-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>210-240</td>
<td>350-380</td>
</tr>
<tr>
<td>2</td>
<td>80-90</td>
<td>110-120</td>
</tr>
<tr>
<td>3</td>
<td>90-110</td>
<td>140-150</td>
</tr>
<tr>
<td>4</td>
<td>75-85</td>
<td>100-105</td>
</tr>
<tr>
<td>5</td>
<td>150-170</td>
<td>210-240</td>
</tr>
<tr>
<td>6</td>
<td>90-110</td>
<td>140-160</td>
</tr>
</tbody>
</table>

According to the constraints considered in this work among inequality constraints only active power constraints are considered. There operating limit of maximum and minimum power are also different. The unit operating ranges are:

- 100 MW ≤ P_1 ≤ 500 MW
- 50 MW ≤ P_2 ≤ 150 MW
- 50 MW ≤ P_3 ≤ 200 MW
- 50 MW ≤ P_4 ≤ 120 MW
- 80 MW ≤ P_5 ≤ 300 MW
- 50 MW ≤ P_6 ≤ 200 MW

The transmission line losses can be calculated by knowing the loss coefficient. The B_{ij} Coefficient matrix is given by

\[
B_{ij} = \begin{bmatrix}
0.0224 & 0.103 & 0.0016 & -0.0053 & 0.0009 & -0.0013 \\
0.0103 & 0.0158 & 0.0010 & -0.074 & 0.0007 & 0.0024 \\
0.0016 & 0.0010 & 0.0474 & -0.0 & -0.0060 & -0.0350 \\
-0.0053 & -0.0074 & -0.0687 & 0.3464 & 0.0105 & 0.0534 \\
0.0009 & 0.0007 & -0.0060 & 0.0105 & 0.0119 & 0.0007 \\
-0.0013 & 0.0024 & -0.0350 & 0.0534 & 0.0007 & 0.2353 \\
\end{bmatrix}
\]

\[B_{ij} = \begin{bmatrix}
-0.0005 & 0.0016 & -0.0029 & 0.0060 & 0.0014 & 0.0015 \\
\end{bmatrix} \text{ and } B_{ij} = 0.0011
\]

6.2 PSO method

The Parameters c1, c2 and inertia weight are selected for best convergence characteristic. Here, c1 = 2.0 and c2 = 2.0. Here the maximum value of w is chosen 0.9 and minimum value is chosen 0.4. The velocity limits are selected as \( V_{max} = 0.5 \cdot P_{max} \) and the minimum velocity is selected as \( V_{min} = 0.5 \cdot P_{min} \).

Table 3: Six Unit System by PSO method

<table>
<thead>
<tr>
<th>Power demand (MW)</th>
<th>500</th>
<th>900</th>
<th>1100</th>
<th>1236</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>223.75</td>
<td>355.56</td>
<td>411.99</td>
<td>473.97</td>
</tr>
<tr>
<td>P2</td>
<td>50</td>
<td>67.768</td>
<td>88.237</td>
<td>117.23</td>
</tr>
<tr>
<td>P3</td>
<td>136.91</td>
<td>171.28</td>
<td>181.74</td>
<td>181.74</td>
</tr>
<tr>
<td>P4</td>
<td>50</td>
<td>110.17</td>
<td>57.403</td>
<td>64.419</td>
</tr>
<tr>
<td>P5</td>
<td>80.55</td>
<td>203.19</td>
<td>255.24</td>
<td>287.83</td>
</tr>
<tr>
<td>P6</td>
<td>50</td>
<td>93.85</td>
<td>129.41</td>
<td>157.02</td>
</tr>
<tr>
<td>Power Loss</td>
<td>3.80</td>
<td>8.94</td>
<td>13.56</td>
<td>19.21</td>
</tr>
<tr>
<td>Total Gen cost (RS/Hr)</td>
<td>6114.8</td>
<td>10709</td>
<td>13268</td>
<td>15477</td>
</tr>
<tr>
<td>Elapsed time (sec)</td>
<td>10.057</td>
<td>10.048</td>
<td>10.408</td>
<td>10.520</td>
</tr>
</tbody>
</table>

Figure 2: Cost curve of 900MW demand by PSO method

Figure 3: Cost curve of 1263 MW demand by PSO method

6.3. BAT Algorithm method

The parameters of algorithm used for simulation are selected for best convergence characteristic. Here, \( n = 100 \) (size of population), \( N_{gen}=500 \) (no of iterations), \( A=0.9 \) (loudness), \( r=0.1 \) (rate of pulse emission), \( Q_{min} = 0 \) and \( Q_{max} = 2 \) (frequency). There are 100 no of bats are selected in the population. For different value of \( n \) and \( N_{gen} \), the cost curve converges in the different region. So, the best value is taken for the minimum cost of the problem. If the no of bats are increased in population then cost curve converges faster. It can be observed the loss has no effect on the cost characteristic.
Table 4: six unit system by bat method

<table>
<thead>
<tr>
<th>Power demand (MW)</th>
<th>500</th>
<th>900</th>
<th>1100</th>
<th>1236</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>139.14</td>
<td>265.91</td>
<td>475.81</td>
<td>500</td>
</tr>
<tr>
<td>P2</td>
<td>69.67</td>
<td>61.513</td>
<td>110.07</td>
<td>150</td>
</tr>
<tr>
<td>P3</td>
<td>52.228</td>
<td>57.666</td>
<td>120.97</td>
<td>200</td>
</tr>
<tr>
<td>P4</td>
<td>133.94</td>
<td>163.26</td>
<td>153.34</td>
<td>164.54</td>
</tr>
<tr>
<td>P5</td>
<td>86.649</td>
<td>219.47</td>
<td>155.49</td>
<td>80</td>
</tr>
<tr>
<td>P6</td>
<td>142.08</td>
<td>151.31</td>
<td>105.8</td>
<td>200</td>
</tr>
<tr>
<td>Power Loss</td>
<td>13.15</td>
<td>19.78</td>
<td>21.05</td>
<td>31.30</td>
</tr>
<tr>
<td>Total Gen cost (RS/Hr)</td>
<td>2443</td>
<td>11186</td>
<td>13609</td>
<td>15472</td>
</tr>
<tr>
<td>Elapsed time (sec)</td>
<td>7.68</td>
<td>7.65</td>
<td>7.96</td>
<td>8.15</td>
</tr>
</tbody>
</table>

6.4 IWD Method

Table 5: six unit system by IWD method [16]

<table>
<thead>
<tr>
<th>Power demand (MW)</th>
<th>1236</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>447.49</td>
</tr>
<tr>
<td>P2</td>
<td>173.32</td>
</tr>
<tr>
<td>P3</td>
<td>263.47</td>
</tr>
<tr>
<td>P4</td>
<td>139.09</td>
</tr>
<tr>
<td>P5</td>
<td>165.47</td>
</tr>
<tr>
<td>P6</td>
<td>87.12</td>
</tr>
<tr>
<td>Power Loss</td>
<td>12.98</td>
</tr>
<tr>
<td>Total Gen cost (RS/Hr)</td>
<td>15450</td>
</tr>
<tr>
<td>Elapsed time (sec)</td>
<td>10.79</td>
</tr>
</tbody>
</table>

Figure 4: Cost curve of 900MW demand by BAT Algorithm method

Figure 5: Cost curve of 1100MW demand by BAT Algorithm method

Figure 6: Cost curve of 1263MW demand by BAT Algorithm method

Figure 7: Cost curve of 1263MW demand by IWD method

6.5. Comparison of methods at 1236 mw

Table 6: comparison of three methods

<table>
<thead>
<tr>
<th>Power demand (MW)</th>
<th>BAT</th>
<th>PSO</th>
<th>IWD</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>500</td>
<td>473.97</td>
<td>447.49</td>
</tr>
<tr>
<td>P2</td>
<td>150</td>
<td>117.23</td>
<td>173.32</td>
</tr>
<tr>
<td>P3</td>
<td>200</td>
<td>181.74</td>
<td>263.47</td>
</tr>
<tr>
<td>P4</td>
<td>164.54</td>
<td>64.419</td>
<td>139.09</td>
</tr>
<tr>
<td>P5</td>
<td>80</td>
<td>287.83</td>
<td>165.47</td>
</tr>
<tr>
<td>P6</td>
<td>200</td>
<td>157.02</td>
<td>87.12</td>
</tr>
<tr>
<td>Power Loss</td>
<td>31.30</td>
<td>19.21</td>
<td>12.98</td>
</tr>
<tr>
<td>Total Gen cost (RS/Hr)</td>
<td>16012</td>
<td>15477</td>
<td>15450</td>
</tr>
<tr>
<td>Elapsed time (sec)</td>
<td>8.15</td>
<td>10.520</td>
<td>10.79</td>
</tr>
</tbody>
</table>

It has been observed that based on results obtained from table 6, The BAT algorithm is very efficient and takes very less time to convergence around 8.15 sec where as in PSO takes around 10.52 sec to converge. Bat algorithm has been compared with the PSO, IWD algorithms and it has been found that, it gives minimum cost of 16,012 Rs/hrs with minimum loss of 31.30 MW at 1263 power demand. The convergence characteristic of the BAT method for 6-unit system is shown in Figs (1-7). All the methods give the minimum cost are not always equal. The performance depends on randomly generated particle in PSO, The static and dynamic parameters in IWD and bats movements in BAT ALGORITHM. Sometimes PSO gives better result and sometimes BAT gives better result. But convergence is
good in BAT ALGORITHM compare to remaining exist methods.

7. Conclusion and Future Scope

The most economical operation of modern power systems is allocate the optimal power generation from different units at the lowest cost possible while meeting all system constraints. Economic Load Dispatch (ELD) is a method to schedule the power generator outputs with respect to the load demands. Economic Load dispatch problem here solved for a six thermal unit system with only transmission losses considered. The three different methods are performing in the MATLAB environment. The problem of six units system when transmission losses are solved by three different methods. In conventional methods better cost is obtained but the problem converges when parameters varied randomly. The cost characteristic takes many numbers of iterations to converge. In BAT ALGORITHM the cost characteristic converges in less elapsed time when compared with PSO and IWD methods. In BAT ALGORITHM the selection of parameters are very important. The best results were obtained when number of population is decreased. In BAT ALGORITHM method selection of parameters are important. So, the parameters may be optimized by using the ANN method. Any other method can be applied with PSO to improve the performance of the BAT method. Here the loss co-efficient are given in the problem. The work may be extended for the problem where transmission loss co-efficient are not given. In that case the loss co-efficient can be calculated by solving the load flow problem.

References

[2] Saumendra Sarangia a thesis on “particle swarm optimization applied to Economic load dispatch”, NIT

Authors Profile

Mr. Ramesh Chevuru received the B.Tech degree in Electrical and Electronics Engineering from Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India and he is currently pursuing M.Tech in the same Institution. He had 6 years experience in teaching as Assistant professor in C.R.E. College, Tirupati, Andhra Pradesh.

Dr. G. V. Marutheswar received the B.Tech, M.Tech and PHD degrees from Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India. He is currently working as professor at EEE Department in Sri Venkateswara University College of Engineering.