A Gravitational Search Algorithm for Solving Economic Load Dispatch Problem

S. Khandualo, A. K Barisal, P. K Pradhan, P. K. Patro
Asst Manager (El) CHEP, Elect Engg Dept., Mechanical Engg Dept,
Chiplima, OHPC Ltd, VSSUT,Burla, VSSUT,Burla,
Dist-Sambalpur,Odisha, India, Dist-Sambalpur,Odisha, India

Abstract—This paper presents the application of a new optimization algorithm i.e Gravitational Search Algorithm (GSA) based on the law of gravity and mass interactions for solving the Economic load dispatch problem of a power system. The Economic load dispatch is the dispatch of available electricity generation resources to supply the load and losses in the transmission links in such a manner that total cost of production of thermal generation is minimized satisfying the constraints in the system. The GSA technique is applied to a six generator twenty-six bus test system and a twenty generator test system to illustrate the effectiveness of the proposed algorithm. Numerical results show that the proposed algorithm is capable of finding very nearly global solutions and achieves cheaper generation schedule in comparison to the other published methods.

Keywords—Economic Dispatch, Gravitational Search Algorithm, Prohibited Operating Zones

I. INTRODUCTION (Heading 1)

Economic load dispatch has become a vital task for proper operation and planning of power system. The main objective of Economic Load Dispatch problem is to minimize the total system operating cost represented by the fuel cost required for the system thermal generation while satisfying all units and system operational constraints i.e equality and inequality constraints such as load balance constraint, ramp rate limits, multi fuel options, prohibited operation zones etc. Conventional methods based on Lagrangian multiplier, gradient search techniques[1], Dynamic Programming[1] require models of thermal plants to be represented as piecewise linear or polynomial approximations of monotonically increasing nature. But such an approximation may lead to suboptimal solution resulting in huge loss of revenue over the time. Methods based on Dynamic Programming, Lambda Iteration methods, Gradient Search methods[1] to solve the Economic Load Dispatch problem was found that it provides solution but it will fail to obtaining solution feasibility and become more complex. Stochastic search algorithms like Tabu Search[2], Genetic Algorithm[3,8,12], Evolutionary Programming[5], Particle Swarm Optimization[7] have been proved to be very exciting in solving complex power systems problems, but these heuristic methods do not always guarantee the globally optimal solution. In this paper a new population based search algorithm called Gravitational search algorithm [6,10] is applied to two different test systems, Test System-I adopted from [12] and Test system-II adopted from [4] and the simulation results are compared to that of Bio geography based optimization [11], Hopfield Modeling framework [4], and Lambda Iteration Method [4], Intelligent Water Drop Algorithms[12], etc. The results show the superiority of the proposed algorithm in solving the complex optimization problem in terms of minimization of cost, minimization of loss and computational time.

II. ECONOMIC LOAD DISPATCH PROBLEM FORMULATION

The main objective of Economic load dispatch problem is to minimize the total generation cost by economic loading of generators such that the operational and network constraints are satisfied.

Objective function is to minimize

$$F_{total} = \sum_{i=1}^{m} F_i(P_i)$$ \hspace{1cm} (1)

The cost function of \textit{i}th unit \(F_i(P_i)\) is a quadratic polynomial and is expressed as

$$F_i(P_i) = a_i + b_i(P_i) + c_i(P_i^2)$$ \hspace{1cm} (2)

where \(a_i, b_i, \text{ and } c_i\) are fuel cost coefficients of \textit{i}th unit and \(m\) is the total number of committed units.

The solution to the problem must satisfy the operational and network constraints in the system.

The constraints are given below:
A. Power Balance Constraint

The total generation \( \sum_{i=1}^{m} P_i \) should be equal to the total system demand \( P_D \) and total transmission loss \( P_{loss} \), i.e

\[
\sum_{i=1}^{m} (P_i) = P_D + P_{loss} ............................................ (3)
\]

The transmission loss is represented as

\[
P_L = \sum_{i=1}^{m} \sum_{j=1}^{m} P_i B_{ij} P_j + \sum_{i=1}^{m} P_i B_{io} + B_{oo} ...... (4)
\]

\( B_{ij} \): The Transmission loss coefficient.

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

B. Generator operating limits

The output generation of each unit must be within minimum and maximum limit of generation. The optimized result must satisfy the following inequality constraint i.e.

\[
P_{i,\min} \leq P_i \leq P_{i,\max} ..................................................(5)
\]

\( P_i \) is the power output of \( i^{th} \) unit. \( P_{i,\min} \) and \( P_{i,\max} \) are the minimum and maximum real power output of \( i^{th} \) generating unit.

The cost function \( F_i(P_i) \) becomes discontinuous when following factors are considered

C. Valve point loadings

Large steam turbine generators are having a number of steam admission valves that are opened in sequence to obtain ever increasing output of the unit. When the valve is first opened, the throttling losses increase rapidly and the incremental heat rate rises suddenly. This causes a ripple in the heat rate curves, the cost function is no longer a quadratic function but a combination of sinusoidal function and quadratic function i.e represented as

\[
F_i(P_i) = a_i + b_i(P_i) + c_i(P_i^2) + \left[ e_i \times \sin(f_i \times (P_{i,\min} - P_i)) \right] ............................................ (6)
\]

D. Prohibited Operating Zones

Prohibited operating zone means the unit is prohibited from generation due to some technical fault in the machine such as vibration in the shaft bearing, or steam valve operation etc. With prohibited zones, the unit has a fuel cost curve of discontinuous in nature [12].

The additional constraints for Units with prohibited operating zones are

\[
P_{i,j-1} \leq P_i \leq P_{i,j} , j = 2,3,...,n_i ...........................................(8)
\]

\[
P_{i,j} \leq P_i \leq P_{i,\max} ....................................................(9)
\]

Where \( j \) is the number of prohibited operating zones of unit \( i \), \( P_{i,j} \) is the lower limit of \( j^{th} \) prohibited unit. And \( P_{i,j+1} \) is the upper limit of \( (j-1)^{th} \) prohibited operating zone of \( i^{th} \) unit. \( n_i \) is the total number of prohibited operating zone of unit \( i \).

III. GRAVITATIONAL SEARCH ALGORITHM

A. Abbreviations and Acronyms

GSA is one of the recent additions to heuristic algorithms was developed by Rashedi et al. in 2009 [6]. GSA is followed by the physical law of gravity and the law of motion. In the proposed algorithm, agents are considered as objects and their performance is measured by their masses. All these objects attract each other by the gravity force and this force causes a global movement of all objects towards the objects with heavier masses. In the proposed algorithm, agents are considered as objects and their performance is measured by their masses. All these objects attract each other by the gravity force and this force causes a global movement of all objects towards the objects with heavier masses.

Consider a system of ‘m’ masses, in which the position of the ith mass by

\[
P_i = (P_{i,1},...,P_{i,d},...,P_{i,m}), i = 1,2,...,m \quad ... \quad ... (10)
\]

Where \( P_{i,d} \) represents the position of ith mass in the dth dimension. At a specific time ‘t’, a gravitational force on mass ‘i’ from mass ‘j’ is given by

\[
F_{ij}(t) = G(t) \left( \frac{M_{pi}(t) \times M_{pj}(t)}{(R_{ij}(t) + \varepsilon)} \right) (P_{j,d} - P_{i,d}) \quad ... (11)
\]

Where \( M_{pi} \) is the passive gravitational mass related to agent \( i \), \( M_{aj} \) is the active gravitational mass related to agent.
The gravitational constant \( G \) is the Gravitational constant at time \( t \), \( \epsilon \) is a small constant and \( R_{ij}(t) \) is Euclidian distance between two agents \( i \) and \( j \).

\[
R_{ij}(t) = ||P_i(t) - P_j(t)||_2
\]

The total force acting on the agent \( i \) in the dimension \( d \) is calculated as follows

\[
F_i(d)(t) = \sum_{j=1, j \neq i}^{m} r_{ij} F_j(d)(t)
\]

Where \( r_{ij} \) is a random number in the interval \([0,1]\).

The acceleration of the agent \( i \) at time \( t \), and in direction \( d \)th 

\[
a_i^d(t) = F_i^d(t)/M_i(t),
\]

\( M_i \) is the inertia mass of \( i \)th agent.

The velocity of a particle is a function of its current velocity added to its current acceleration. Therefore the next position and next velocity of an agent can be calculated as

\[
v_i^d(t+1) = v_i^d(t) + a_i^d(t)
\]

\[
p_i^d(t+1) = p_i^d(t) + v_i^d(t+1)
\]

The Gravitational constant \( G \) is initialized at the beginning and will be decreased with time to control the search accuracy, i.e \( G \) is a function of the initial value \( G_0 \) and time \( t \).

\[
G(t) = G_0 \alpha^{-t/\text{iteration max number iterations}}
\]

\( \alpha \) is a constant, iteration is the current iteration and maximum iteration is the maximum number of iterations given.

The masses of the agents are calculated using fitness evaluation. A heavier mass means a more efficient agent. This means that better agents have higher attractions and moves more slowly.

\( fit_i(t) \) is the fitness value of agent \( i \) at time \( t \), \( best(t) \) and \( worst(t) \) represents the strongest and weakest agent according to their fitness value.

For a minimization problem, \( best(t) = \min_{j \in \{1, ..., m\}} fit_j(t) \)

and \( worst(t) = \max_{j \in \{1, ..., m\}} fit_j(t) \)

The gravitational and inertial masses are updating by the following equations

\[
m_i(t) = fit_i(t) - worst(t)/best(t) - worst(t)
\]

\[
M_i(t) = m_i(t)/\sum_{j=1}^{m} m_j(t)
\]

IV. RESULTS AND DISCUSSION

The proposed gravitational search algorithm was tested on two test systems having six units with 26 bus test system and twenty unit test systems. The GSA was programmed in MATLAB environment and executed on a 2.30GHz Pentium-III processor with 4GB RAM. The simulation results were compared with other published work, result reveals the superiority of the proposed algorithm.

A. Test System-I

The system contains six thermal units, 26 buses, and 46 transmission lines [12]. The load demand is 1263MW. The load balance constraint, generation limit constraint and prohibited operating zone constraints are considered in the system. The input data for test system-I are furnished in Table-I, Table-II and the B-Coefficients taken from [12].

B. Test System-II:

The Test System-II consists of Twenty Thermal units and supplies a total load demand of \( P_D = 2500 \)MW, B coefficient matrix is adopted from [4].

### Table I. Input Data of Test System-I

<table>
<thead>
<tr>
<th>Unit</th>
<th>( a_i ) ($/h)</th>
<th>( b_i ) ($/h)</th>
<th>( c_i ) ($/h)</th>
<th>( P_{min} ) (MW)</th>
<th>( P_{max} ) (MW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1000</td>
<td>18.19</td>
<td>0.00068</td>
<td>150</td>
<td>600</td>
</tr>
<tr>
<td>2</td>
<td>970</td>
<td>19.26</td>
<td>0.00071</td>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>3</td>
<td>600</td>
<td>19.8</td>
<td>0.0065</td>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>4</td>
<td>700</td>
<td>19.1</td>
<td>0.005</td>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>5</td>
<td>420</td>
<td>18.1</td>
<td>0.00738</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>360</td>
<td>19.26</td>
<td>0.00814</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>7</td>
<td>490</td>
<td>17.14</td>
<td>0.0079</td>
<td>25</td>
<td>125</td>
</tr>
<tr>
<td>8</td>
<td>660</td>
<td>18.92</td>
<td>0.00813</td>
<td>50</td>
<td>150</td>
</tr>
<tr>
<td>9</td>
<td>765</td>
<td>18.27</td>
<td>0.00522</td>
<td>50</td>
<td>200</td>
</tr>
<tr>
<td>10</td>
<td>770</td>
<td>18.92</td>
<td>0.00573</td>
<td>30</td>
<td>150</td>
</tr>
<tr>
<td>11</td>
<td>800</td>
<td>16.69</td>
<td>0.0048</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>12</td>
<td>970</td>
<td>16.76</td>
<td>0.00314</td>
<td>150</td>
<td>500</td>
</tr>
<tr>
<td>13</td>
<td>900</td>
<td>17.36</td>
<td>0.0085</td>
<td>40</td>
<td>160</td>
</tr>
<tr>
<td>14</td>
<td>700</td>
<td>18.7</td>
<td>0.00511</td>
<td>20</td>
<td>130</td>
</tr>
<tr>
<td>15</td>
<td>450</td>
<td>18.7</td>
<td>0.00398</td>
<td>25</td>
<td>185</td>
</tr>
<tr>
<td>16</td>
<td>370</td>
<td>14.26</td>
<td>0.0712</td>
<td>20</td>
<td>80</td>
</tr>
<tr>
<td>17</td>
<td>480</td>
<td>19.14</td>
<td>0.0089</td>
<td>30</td>
<td>85</td>
</tr>
<tr>
<td>18</td>
<td>680</td>
<td>18.92</td>
<td>0.00713</td>
<td>30</td>
<td>120</td>
</tr>
<tr>
<td>19</td>
<td>700</td>
<td>18.47</td>
<td>0.00622</td>
<td>40</td>
<td>120</td>
</tr>
<tr>
<td>20</td>
<td>850</td>
<td>19.79</td>
<td>0.00773</td>
<td>30</td>
<td>100</td>
</tr>
</tbody>
</table>
TABLE III. COMPARATIVE RESULTS OF TEST SYSTEM-I

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>474.80</td>
<td>447.497</td>
<td>447.399</td>
<td>450.13</td>
<td>462.6177</td>
</tr>
<tr>
<td>2</td>
<td>178.63</td>
<td>173.322</td>
<td>173.239</td>
<td>173.62</td>
<td>184.9532</td>
</tr>
<tr>
<td>3</td>
<td>262.20</td>
<td>263.474</td>
<td>263.316</td>
<td>260.61</td>
<td>256.6655</td>
</tr>
<tr>
<td>4</td>
<td>134.28</td>
<td>139.059</td>
<td>138.000</td>
<td>139.49</td>
<td>133.0794</td>
</tr>
<tr>
<td>5</td>
<td>151.90</td>
<td>165.476</td>
<td>165.410</td>
<td>159.7</td>
<td>152.3787</td>
</tr>
<tr>
<td>6</td>
<td>74.181</td>
<td>87.128</td>
<td>87.0797</td>
<td>90.51</td>
<td>85.3885</td>
</tr>
<tr>
<td>Total Gen</td>
<td>1276.0</td>
<td>1276.01</td>
<td>1275.44</td>
<td>1274.05</td>
<td>1275.083</td>
</tr>
<tr>
<td>Ploss</td>
<td>13.021</td>
<td>12.9584</td>
<td>12.446</td>
<td>12.05</td>
<td>12.083</td>
</tr>
<tr>
<td>Total Gen Cost ($/h)</td>
<td>15459</td>
<td>15450</td>
<td>15446.0</td>
<td>15439</td>
<td>15373.83</td>
</tr>
<tr>
<td>CPU time per iteration (s)</td>
<td>0.22</td>
<td>0.06</td>
<td>0.0638</td>
<td>0.0254</td>
<td>0.00807</td>
</tr>
</tbody>
</table>

The convergence characteristics of the proposed method on Test system-I and Test system-II are provided in Figs. 1 and 2, respectively. The results obtained by the proposed GSA method on Test System-I are compared with genetic algorithm (GA), particle swarm optimization (PSO), biogeography based optimization (BBO) and intelligent water drop algorithm (IWD) methods in Table III. The GSA method provides cheaper generation schedule in comparison to all other above methods in less execution time. Similarly in Table IV, the proposed GSA outperforms to other established methods in terms of quality of solution and execution time. Moreover, GSA method performs better to other methods reported in literature in both of the test systems considered for study.

TABLE IV. OPTIMAL SOLUTION OF TEST SYSTEM-II

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>512.780</td>
<td>512.7804</td>
<td>513.089</td>
<td>570.7216</td>
</tr>
<tr>
<td>2</td>
<td>169.103</td>
<td>169.1035</td>
<td>173.353</td>
<td>174.6018</td>
</tr>
<tr>
<td>3</td>
<td>126.889</td>
<td>126.8897</td>
<td>126.923</td>
<td>98.8116</td>
</tr>
<tr>
<td>4</td>
<td>102.865</td>
<td>102.8656</td>
<td>103.329</td>
<td>87.7545</td>
</tr>
<tr>
<td>5</td>
<td>113.683</td>
<td>113.6836</td>
<td>113.774</td>
<td>123.2827</td>
</tr>
<tr>
<td>6</td>
<td>73.571</td>
<td>73.5709</td>
<td>73.0669</td>
<td>71.0592</td>
</tr>
<tr>
<td>7</td>
<td>115.287</td>
<td>115.2876</td>
<td>114.984</td>
<td>99.7516</td>
</tr>
<tr>
<td>8</td>
<td>116.399</td>
<td>116.3994</td>
<td>116.423</td>
<td>98</td>
</tr>
<tr>
<td>9</td>
<td>100.406</td>
<td>100.4063</td>
<td>100.694</td>
<td>122.1304</td>
</tr>
<tr>
<td>10</td>
<td>106.026</td>
<td>106.0267</td>
<td>99.9997</td>
<td>106.9398</td>
</tr>
<tr>
<td>11</td>
<td>150.239</td>
<td>150.2395</td>
<td>148.977</td>
<td>154.3317</td>
</tr>
<tr>
<td>12</td>
<td>292.764</td>
<td>292.7647</td>
<td>294.020</td>
<td>298.2807</td>
</tr>
<tr>
<td>13</td>
<td>119.115</td>
<td>119.1155</td>
<td>119.575</td>
<td>120.3959</td>
</tr>
<tr>
<td>14</td>
<td>30.834</td>
<td>30.8342</td>
<td>30.5478</td>
<td>29.8224</td>
</tr>
<tr>
<td>15</td>
<td>115.805</td>
<td>115.8056</td>
<td>116.454</td>
<td>115.6996</td>
</tr>
<tr>
<td>16</td>
<td>36.2545</td>
<td>36.25455</td>
<td>36.2278</td>
<td>37</td>
</tr>
<tr>
<td>17</td>
<td>66.859</td>
<td>66.8594</td>
<td>66.8594</td>
<td>61.7785</td>
</tr>
<tr>
<td>18</td>
<td>87.972</td>
<td>87.972</td>
<td>88.5470</td>
<td>85.0294</td>
</tr>
<tr>
<td>19</td>
<td>100.803</td>
<td>100.8033</td>
<td>100.980</td>
<td>84.2168</td>
</tr>
<tr>
<td>20</td>
<td>54.305</td>
<td>54.305</td>
<td>54.2725</td>
<td>51.4542</td>
</tr>
<tr>
<td>Total Gen</td>
<td>2591.96</td>
<td>2591.967</td>
<td>2592.10</td>
<td>2591.062</td>
</tr>
<tr>
<td>Ploss</td>
<td>91.967</td>
<td>91.9669</td>
<td>92.101</td>
<td>91.0624</td>
</tr>
<tr>
<td>Total Gen Cost ($/h)</td>
<td>62456.6</td>
<td>62456.63</td>
<td>62456.7</td>
<td>62314.94</td>
</tr>
<tr>
<td>CPU time in Sec</td>
<td>33.757</td>
<td>6.355</td>
<td>0.02928</td>
<td>2.525</td>
</tr>
</tbody>
</table>

The convergence characteristics of the proposed method on Test system-I and Test system-II are provided in Figs. 1 and 2, respectively. The results obtained by the proposed GSA method on Test System-I are compared with genetic algorithm (GA), particle swarm optimization (PSO), biogeography based optimization (BBO) and intelligent water drop algorithm (IWD) methods in Table III. The GSA method provides cheaper generation schedule in comparison to all other above methods in less execution time. Similarly in Table IV, the proposed GSA outperforms to other established methods in terms of quality of solution and execution time. Moreover, GSA method performs better to other methods reported in literature in both of the test systems considered for study.

Fig.1: Convergence characteristics of GSA on Test System-I

Fig.2: Convergence characteristics of GSA on Test System-II
V. CONCLUSION

A new algorithm called Gravitational Search Algorithm was developed and demonstrated on two test systems to solve the economic load dispatch of two different test systems. Results show that GSA based algorithms are more capable of finding highly near-global solutions than lamda iteration method, Hopfield model, BBO etc. The optimal cost obtained by the GSA is quite cheaper than the other published work for the system adopted. In future, attempts can be made to apply the hybrid gravitational search algorithm to large thermal system in conjunction with hydro, wind energy by incorporating emission, spinning reserve and reliability constraints.

REFERENCES