Neuro Inspired Genetic Hybrid Algorithm for Active Power Dispatch Planning Problem in Small Scale System

Navpreet Singh Tung\textsuperscript{1} and Prof. (Dr). Sandeep Chakravorty\textsuperscript{2}

\textsuperscript{1}Assistant Professor, Department of Electrical Engineering, Bhutta Group of Institutions, Ludhiana India
\textsuperscript{2}Dean and Professor, Department of Electrical Engineering, Baddi University, Baddi, India
\textsuperscript{1}icenitj@gmail.com, \textsuperscript{2}sandeep@baddiuniversity.ac.in

Abstract

Allocation of optimum active power is a backbone of power system generation planning and its high impact contribution is the need of current electrical utilities and power engineers need to browse this area in short and long term planning scenarios. Power demand requirements mapped to economic feasible solutions matching voltage profile, power demand, minimization of losses, voltage stability and improve the capacity of the system is the need of the hour. Modern techniques based on evolutionary computing, artificial intelligence, search method find their objectives in the area of economic load dispatch planning to reach global optimal solution for this multi-decision, multi-objective combinatorial problem subjected to different constraints. Many algorithms suffer from global convergence problem. To vanish this drawback, neuro inspired genetic hybrid algorithm (NIGHA) has been proposed in this paper to solve economic dispatch problem. Unlike other algorithms, NIGHA utilizes the weights of Neural Network to explore information and knowledge to train GA parameters to search for feasible region where optimal global solution converges. The suggested technique is tested on IEEE 25 bus system. Test results are compared with other techniques presented in literature. Proposed technique has outperformed other methods in terms of cost, computation time.

Keywords: Neuro Inspired Genetic Hybrid Algorithm(NIGHA) Genetic Algorithm(GA), Economic Dispatch (ED), Neural Network(NN)

1. Introduction

The economic dispatch (ED) problem is one of the most important areas of today’s power system. The purpose of the ED is to find the optimum generation among the existing units, such that the total generation cost is minimized while simultaneously satisfying the power balance equations and various other constraints in the system. Below are the suggested techniques in the literature -


2. Problem Formulation

The ED problem[45,46,47] may be expressed by minimizing the fuel cost of generator units under constraints. Depending on load variations, the output of generators has to be changed to meet the balance between loads and generation of a power system. The power system model consists of n generating units already connected to the system.

The ED problem can be expressed as.

2.1. Fuel Cost Model

\[ C(P_{Gi}) = \sum (a_i*P_{Gi}^2 + b_i*P_{Gi} + c_i)R_s \]

where \(i=1\ldots N\)

2.2. Constraints

- \(\sum P_{Gi} - P_L + P_D = 0\)
- \(P_{Gi,\min} \leq P_{Ci} \leq P_{Gi,\max}\) where \(i=1,2\ldots N\)

2.3. Minimization

Total Operating Cost = \(C\)

2.4. Transmission Losses

\[ P_L = \sum_{i=1}^{N} \sum_{j=1}^{N} P_{ij}B_{ij}P_{ij} + \sum_{i=1}^{N} B_{ii}P_{Gi} + B_{00} \]

3. Proposed Technique

3.1. Genetic Algorithm

Genetic Algorithm (GA)[48] is a search directed algorithm inspired by survival of the fittest among string structures to form a search algorithm. For reliable solution of optimization problems, GA has been investigated robustly and proved to be effective at exploring a complex space in an adaptive way, directed by the biological evolution mechanisms of reproduction, crossover and mutation. GA executes the search process in multiple phases: Initialization, Selection, Crossover and Mutation. GA serves as search optimization technique for adaptation of network weights. It is applied to upgrade the performance of ANN i.e. to check the connection weight in the form of binary or real number. It is also applicable for topology selection, training of network, determining the number of nodes in each layer, evolution of connection weights, evolution of learning rule etc. GA does not offer constraint with scaling as back propagation. The reason is that they
generally improve the current best candidate monotonically. They perform this by keeping
the current best individual as part of their population while they search for better
candidates. GA generally does not bothered by local minima. The mutation and crossover
operators can step from a peak across a hill to an even lower peak with no more difficulty
than jump directly into a peak.

3.2. Flow Chart of GA

3.3. Artificial Neural Network

ANN, [48, 49] often called a artificial neural network, is a mathematical model inspired
by biological neural networks. The motivation for the development of neural network
technology originated from the zeal to implement an artificial system that could perform
intelligent function intersect to those performed by the human brain. It is a powerful data
modeling tool that is able to store and show complex input output relationships. In most
cases a neural network is an adaptive system, between inputs and outputs, to find patterns
in data. Neural network copy the human brain in the following two ways:
1) A neural network acquires its knowledge through learning.
2) A neural network’s knowledge is stored within interneuron connection strengths known
as synaptic weights. A neural network comprises of an interconnected group of artificial
neurons, and it processes information using a connectionist approach to computation. An
ANN is composed [48, 49] of neurons that are processing elements in a network. Each
neuron gets input data, processes it and gives a single output. The input data can be raw
data or output of other processing elements. The output can be the output or it can be an
input to another neuron. An ANN is formed by nodes engaged together. Nodes with
similar characteristics are arranged into layer. A layer can be seen as a group of nodes
which have connections to other layers, or to external environment, but which have no
interconnections. There are basically three types of layers. The first layer connecting to
the input variables is called input layer. The last layer connecting to the output variables is
called the output layer. It is straightforward to extend to multiple nodes. Layers between
the input and output layers are called hidden layers. Information is transmitted through the
connections between nodes. This type of network is called feed forward network, or multilayer feed forward network. Layered Feed forward networks have been explored lot. First they have been used to generalize well. Secondly, a training algorithm called back propagation happen which can often find a good set of weights (and biases) in a reasonable amount of tune. Back propagation is a variation on gradient search. It generally exploits a least square criterion.

3.4. NIGHA

NIGHA, a neuro inspired genetic algorithm is proposed for active power dispatch planning problem. The factors influencing NIGHA evolution are genetic operators, fitness function and stopping criteria. The genetic operators namely; two-point crossover and uniform mutation are applied. The fitness function of NIGHA is chosen to be minimization of cost. Convergence of the network is ensured and computation is halted when the objective variation in iterations is within tolerance. Primarily in GA, a set of initial encoded schedules known as chromosomes is randomly created. Each schedule is valuated for "fitness". Then, processes based on natural selection, crossover, and mutation are repeated applied on a population of binary strings which represent potential solutions. Over time, the number of above–average individuals increases and better–fit individuals are created, until a good solution to the problem at hand is found [49].

3.4.1. Algorithm

1. Initialize the weights and bias of network
2. Create input and target vectors to suit the network
3. Map chromosomes from the population to the network
4. Run and simulate network
5. Test the fitness function
6. Increase the number of hidden neurons and repeat the process, until minimum objective function value achieved
7. Do three point crossover on the population to exchange information between parents
8. Get the best population, choose parent and mutate
9. Stop if the condition is satisfied

3.4.2 Pseudo code of NIGHA

```
{ Initialize i ← 0; j ← 0(iteration);
  Generate the initial population P_i of the real-coded chromosomes C_i;
  W_j - representing a weight set for the NIGHA;
  While (condition-the current population P_i is not converged)
  { Generate fitness values F_i for each C_i using the Algorithm NIGHA ( );
    Fetch the mating pool ready by terminating worst fit individuals and duplicating high fit individuals;
    Using the cross over to reproduce offspring from the parent chromosomes;
    i ← i +1; increment counter
    Invoke the current population P_i;
  }
  Calculate objective function values F_i for each C_i
}
```
3.4.3. Flow Chart of NIGHA

4. Active Power Dispatch using NIGHA
   - Variables
     Power Generation (PG) and cost coefficients (a,b,c) of units with objective function as fuel cost, quadratic in nature. Power Generation variable should be initialized as starting point for initial solution in ant bee colony algorithm.
   - Constraints
     Equality Constraints: Power Generation-Power Demand-Power losses=0(P_G-P_d-P_L)

© 2015 SERSC
In-Equality Constraints: Power Generation should be between minimum and maximum limit of power generation. 
Variables in constraints should be incorporated in pattern search algorithm.

- Stopping Criteria
It is maximum generation limit for optimum solution.

5. Simulation Results
This proposed approach is tested on IEEE 25 bus system [44, 45, 46]. Simulation results are achieved and compared with other techniques presented in literature. In Table 1 and Table 2, total cost and computational time has been evaluated using NIGHA. Optimal active power generation on IEEE 25 bus system have been presented. Comparison with other techniques like differential evolution, pattern search, real and binary coded genetic algorithm have been carried out. In Table 3, parameter setup used in NIGHA have been stored. Performance evaluation of NIGHA on training, test data has been carried out.

Table 1. Power Generation, Total Cost and Computational Time using NIGHA

<table>
<thead>
<tr>
<th>Parameters</th>
<th>NIGHA</th>
<th>DE</th>
<th>PS</th>
<th>RCGAs</th>
<th>BCGAs</th>
<th>BFGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG1 (MW)</td>
<td>268.3668</td>
<td>212.244</td>
<td>212.244</td>
<td>213.68</td>
<td>206.72</td>
<td>211.30</td>
</tr>
<tr>
<td>PG2 (MW)</td>
<td>123.2338</td>
<td>122.789</td>
<td>122.789</td>
<td>127.46</td>
<td>121.64</td>
<td>126.30</td>
</tr>
<tr>
<td>PG3 (MW)</td>
<td>141.4179</td>
<td>140.305</td>
<td>140.305</td>
<td>141.93</td>
<td>151.82</td>
<td>151.29</td>
</tr>
<tr>
<td>PG4 (MW)</td>
<td>27.4627</td>
<td>27.296</td>
<td>27.296</td>
<td>29.53</td>
<td>33.21</td>
<td>71.24</td>
</tr>
<tr>
<td>PG5 (MW)</td>
<td>224.7512</td>
<td>268.366</td>
<td>268.366</td>
<td>258.86</td>
<td>258.05</td>
<td>211.31</td>
</tr>
<tr>
<td>Cost ($/hr)</td>
<td>1921.4</td>
<td>2009.312&lt;sup&gt;4&lt;/sup&gt;</td>
<td>2009.312&lt;sup&gt;3&lt;/sup&gt;</td>
<td>2010.8</td>
<td>2011.0</td>
<td>2029.3</td>
</tr>
<tr>
<td>Time (Sec)</td>
<td>1.1</td>
<td>7.0</td>
<td>1.2</td>
<td>1.6</td>
<td>4.78</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Table 2. Results Comparison with other Techniques [44,45,46]

<table>
<thead>
<tr>
<th>Parameters</th>
<th>NIGHA</th>
<th>DE</th>
<th>PS</th>
<th>RCGAs</th>
<th>BCGAs</th>
<th>BFGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG1 (MW)</td>
<td>268.3668</td>
<td>212.244</td>
<td>212.244</td>
<td>213.68</td>
<td>206.72</td>
<td>211.30</td>
</tr>
<tr>
<td>PG2 (MW)</td>
<td>123.2338</td>
<td>122.789</td>
<td>122.789</td>
<td>127.46</td>
<td>121.64</td>
<td>126.30</td>
</tr>
<tr>
<td>PG3 (MW)</td>
<td>141.4179</td>
<td>140.305</td>
<td>140.305</td>
<td>141.93</td>
<td>151.82</td>
<td>151.29</td>
</tr>
<tr>
<td>PG4 (MW)</td>
<td>27.4627</td>
<td>27.296</td>
<td>27.296</td>
<td>29.53</td>
<td>33.21</td>
<td>71.24</td>
</tr>
<tr>
<td>PG5 (MW)</td>
<td>224.7512</td>
<td>268.366</td>
<td>268.366</td>
<td>258.86</td>
<td>258.05</td>
<td>211.31</td>
</tr>
<tr>
<td>Cost ($/hr)</td>
<td>1921.4</td>
<td>2009.312&lt;sup&gt;4&lt;/sup&gt;</td>
<td>2009.312&lt;sup&gt;3&lt;/sup&gt;</td>
<td>2010.8</td>
<td>2011.0</td>
<td>2029.3</td>
</tr>
<tr>
<td>Time (Sec)</td>
<td>1.1</td>
<td>7.0</td>
<td>1.2</td>
<td>1.6</td>
<td>4.78</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Table 3. Performance Evaluation on Test and Test Data

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training</th>
<th>Test</th>
<th>Val</th>
<th>Actual</th>
<th>Iterations</th>
<th>Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1.6357</td>
<td>1.6352</td>
<td>1.6291</td>
<td>1.6333</td>
<td>220</td>
<td>60</td>
</tr>
</tbody>
</table>

Figure 1. Power Generation Comparison
In Figure 1, Figure 2 and Figure 3 Variation of operating cost, computation time and power generation using NIGHA and other techniques pattern search, differential evolution, genetic algorithms have been shown.

In Figure 4, Search convergence of NIGHA for best solution in feasible region have been presented. Among different global and local values, it achieves its global minima.
In Figure 4 Regression Analysis of training, test and validation phase of data using NIGHA. Fitness level in terms of regression have been linear for curve fit data.

In Figure 6 variation of solution gradients of network with epochs are reflected. Best Gradient at epoch 38 is achieved.

In Figure 7, Reduction of error with increasing epoch in NIGHA. MSE converges to minimum value at epoch 18.
In Figure 8 and Figure 9, histogram representation of errors (20 bins) and error analysis of training, test and valid data is delivered.
In Figure 10 and Figure 11 solution convergences analysis using NIGHA with iteration and comparison with other techniques have been incorporated.

6. Conclusion

An application of soft computing techniques in economic load dispatch planning optimization has been inherently evolving for last few decades. Different evolutionary and intelligent computation methods whether stand alone or hybrid in nature have been developed and successfully applied to economic load dispatch area. In the current research, an application of NIGHA has been applied successfully for economic active power dispatch problem. Proposed technique is tested on IEEE 25 bus system. Test results reveal the minimum operating cost, optimum power generation and high speed convergence of solution. A comparison has been made other techniques presented in literature. It out-performs other techniques presented in literature in terms of computation speed, fuel cost and power generation. Hence, NIGHA algorithm is more robust and lead to optimal solution in economic active power dispatch problem.

7. Future Scope

Upcoming research involves the expansion of NIGHA lead to the formulation and development of hybrid algorithm based on simulated annealing to polish the best solution search capacity of the proposed technique as well as fast convergence for optimal solution with incorporation of practical constraints.

References


References


NOMENCLATURE

N     Number of units
Pd    Power Demand
Pmax  Maximum limit of Unit
Pmin  Minimum Limit of Unit
Pc    Power Generation
C     Total Cost
Ploss Power Losses
a, b, c Cost Coefficients
B     Loss Coefficients

Authors

Navpreet Singh Tung is serving as an Assistant Professor in Department of Electrical Engineering, Bhutta Group of Institutions. He served as an Assistant Professor in Department of Electrical Engineering, C.T. Group of Institutions, India. He holds his B-Tech in Instrumentation and Control Engineering from National Institute of Technology, Jalandhar. He obtained his M-Tech in Electrical Engineering with specialization in Power System from Lovely Professional University. He is a member of reviewer board of Prestigious International Journals. He authored and co-authored many papers in leading international proceedings and journals in Power System. His area of interest is Power System Planning, Power System Optimization.

Prof.(Dr.) Sandeep Chakravorty is serving as a Dean and Professor in Department of Electrical Engineering, Baddi University, India. He did his BE in Department of Electrical and Electronics Engineering, Sikkim Manipal Institute of Technology, Sikkim and ME in Software Engineering from Birla Institute of Technology, Mesra, Ranchi. He obtained his PHD in Power System Planning from Sikkim Manipal University. He served in different capacities in Sikkim Manipal University, Lovely Professional University. He has a long stint of teaching and research career in Electrical Engineering. He authored and co-authored many research papers in the area of Power system in leading International Journals and Conferences. His area of expertise is Power System Planning, Power system Optimization and application of artificial intelligence in Power System.