Hybrid PSO-DS for non-convex economic dispatch problems

T. ARULDOSS ALBERT VICTOIRE
Department of Electrical Engineering
Karunya Institute of Technology,
Coimbatore- 641114
India

A. EBENEZER JEYAKUMAR
Department of Electrical Engineering
GCT campus, Anna University, India
Coimbatore – 641013
India

Abstract: - This paper presents a novel and efficient method for solving the economic dispatch problem (EDP), based on the Particle Swarm Optimization (PSO) with Direct Search (DS) method incorporated. The heuristic integrates DS method with the PSO and fine tunes every improvement of solution of the PSO run. The PSO is used with a linear inertia weight to facilitate a global and local search as the algorithm proceeds. The optimization procedure based on DS and systematic reduction in search region is found effective in solving various problems in the field of non-linear programming. The Hybrid Particle Swarm optimization and Direct Search (PSODS) is a derivative free optimization technique which produces results quickly and proves itself fit for solving large scale EDPs. To validate the effectiveness of the PSODS method, extensive tests were carried out on two different test cases and compared with the results in the recently available literature. The PSODS method for EDP is shown to be general and provides quality solutions compared to other existing heuristics.

Key-Words: - Particle swarm optimization; Direct search; Evolutionary computation: Heuristics; Economic dispatch

1. Introduction

Minimizing the cost of generation of electric power by assigning different load levels to the generating units to meet the required load demand by satisfying various system constraints is termed as EDP. Traditional dispatch algorithms like Lambda Iteration, gradient methods and dynamic programming developed for solving the EDP cease to be applicable while solving problems with non-monotonically increasing functions [1]. Also traditional methods based on coordination equations are the most suitable and fast converging. But these algorithms will stick at local minima when applied to solve non-convex functions. Traditionally the generating units cost functions are assumed to be convex and their incremental heat rate curves exhibit a monotonically increasing characteristic. But in reality large steam turbines have steam admission valves, which cause discontinuities in the incremental heat rate curves. A combined cycle plant will have generators with piece-wise quadratic cost functions.

Thus an algorithm that does not depend on the fuel cost function has to be introduced. In this regard, solution procedure based on stochastic search techniques were evolved [2]. Some of them are Genetic Algorithm (GA) [3], Simulated Annealing (SA) [4], and Evolutionary programming (EP) [5] and the EDP was already solved using these stochastic optimization techniques. These techniques have their own disadvantages in terms of computation time [1].

Recently hybrid heuristic techniques [6,7] were evolved and applied for various power system problems. Recently PSO was introduced by Dr. Kennedy and Dr. Eberhart [8] as one of the modern heuristic algorithms under the evolutionary techniques and gained lots of attention in various power system applications [8]. PSO shares some of the common features available in other evolutionary techniques, except the selection procedure. Also PSO will not follow survival of the fittest, the principle of other evolutionary techniques. PSO when compared to EP it has very fast converging characteristics. In order to improve the fine tuning ability the linear inertia weight and constriction factor approaches were incorporated with the PSO. But there exists a trade-off between the quality and fine tuning of the solution.

In this paper a new technique based on hybrid particle swarm optimization with direct search method (PSODS), is used to solve the EDP. The main drawback of the PSO is its poor fine tuning of solution while exploring the search space. The particle may reach near the global solution in the middle of the run but may escape in the consecutive runs. To retain the best improvement unless a better improvement in the run is explored, a fine tuning is done in that region. To accomplish this, Direct Search method [9], a derivative free local search algorithm based on systematic reduction of search region is integrated with the PSO. Whenever there is an improvement in the solution of
the PSO run the DS method is applied and fine tuned that solution region.

To validate the performance of the PSODS approach two Economic Dispatch Problems (EDP) with different non-smooth objective functions were taken and the results obtained were compared with those obtained using other techniques.

2. EDP formulation

The classic EDP minimizes the following fuel cost function associated to dispatchable units:

$$F_T = \sum_{i=1}^{N} F_i(P_i)$$  \hspace{1cm} (1)

Subject to the following equality and inequality constraints:

a) Real power balance:

$$\sum_{i=1}^{N} P_i = P_D + P_{Loss}$$  \hspace{1cm} (2)

where, the $P_{Loss}$ is calculated using the B- Matrix loss coefficients and expressed in the quadratic form as given below,

$$P_{Loss} = \sum_{m=1}^{N} \sum_{n=1}^{N} P_m B_{mn} P_n$$  \hspace{1cm} (3)

b) Real power generation limit:

$$P_{i\min} \leq P_i \leq P_{i\max}$$  \hspace{1cm} (4)

where,

$$F_T$$ - Total production cost ($$/hr)

$$F_i(P_i)$$ - Fuel cost function ($$/hr)

$$P_i$$ - Real power output of the $i^{th}$ unit (MW)

$N$ - Number of generating units

$$P_D$$ - Power demand (MW)

$$P_{Loss}$$ - Power Loss (MW)

$B_{mn}$ - Transmission loss coefficients

$$P_{i\min}$$ - Minimum limit of the real power of the $i^{th}$ unit (MW)

$$P_{i\max}$$ - Maximum limit of the real power of the $i^{th}$ unit (MW)

The economic dispatch of generation of real power of the generating units is to be done to the required load demand by satisfying the above constraints. The fuel cost function varies as it depends on the various considerations on the generating units. In this paper the following two different modeling of the fuel cost functions were considered,

a) Fuel cost function considering the valve point loading effects,

b) Fuel cost function with piece-wise quadratic cost function.

3. Particle Swarm Optimization: An Overview [8]

Compared to other evolutionary techniques, the advantages of PSO are that PSO is easy to implement and there are only few parameters to adjust. Similar to other Evolutionary algorithms, PSO must also have a fitness evaluation function that takes the agent’s position and assigns to it a fitness value. For consistency the fitness function is the same as for other Evolutionary algorithms. The position with the highest fitness value in the entire run is called the global best ($G_{best}$). Each agent also keeps track of its highest fitness value. The location of this value is called its personal best ($P_{best}$). Each agent is initialized with a random position and random velocity. The velocity in each of $n$ dimensions is accelerated toward the global best and its own personal best based on the following equation:

$$v_{j}^{t+1} = w v_{j}^{t} + c_1 \text{rand} (P_{best} - s_{j}^{t}) + c_2 \text{rand} (G_{best} - s_{j}^{t})$$  \hspace{1cm} (5)

where,

$w$ : inertia weight

The inertia weight is usually calculated using the following expression,

$$w = w_{\max} - (w_{\max} - w_{\min}) \times \frac{iter}{iter_{\max}}$$  \hspace{1cm} (6)

where,

$w_{\max}$ : initial weight,

$w_{\min}$ : final weight,

$iter$ : current iteration number.

$iter_{\max}$ : maximum iteration number.

$v_{j}^{t}$ : velocity of agent $j$ at iteration $t$.

$c_1, c_2$ : weighting factors.

$rand$ : random number between 0 and 1.

$s_{j}^{t}$ : current position of agent $j$ at iteration $t$.

$P_{best}$ : P best of agent $j$.

$G_{best}$ : G best of the group.
\[ s_{j}^{t+1} = s_{j}^{t} + v_{j}^{t+1} \]  

The pseudo code of the PSO algorithm is as follows,

For each agent
   Initialize searching points and velocities
End

Do
   For each agent
      Evaluate the fitness value of searching points
      If the fitness value is better than the current \( P_{best} \) fitness value
         Set current value as the new \( P_{best} \)
   End
   Choose the agent with the best fitness value of all the agents as \( G_{best} \)
   For each agent
      Calculate agent velocity according to equation (5)
      Update agent position according to equation (7)
   End
While maximum iterations or minimum error criteria is not attained

4. Direct Search Method [9]:

The optimization procedure based on direct search and systematic reduction in search region is found effective in solving various problems in the field of non-linear programming. This procedure handles either inequality or equality constraints or the feasible region that does not have to be convex and no approximations or auxiliary variables required. The most attractive features are the ease of setting up the problem on the computer, speed in obtaining the optimum and reliability of the results. For problems having more than one local optimum, this method is especially useful.

Step 1: Set count \( c \)=1. The best solution vector obtained from the PSO algorithm is used as an initial point \( Q(c) \) and an initial range vector is defined as

\[ R(c) = \psi (P_{max} - P_{min}) \]

Where \( \psi \) – Multiplication factor from 0 to 1.

Step 2: Generate \( N_{s} \) trail solution vectors around \( Q(c) \)

using following relationship,

\[ Q_{i} = Q(c) + R(c) \cdot rand(-0.5,0.5) \]

where \( Q_{i} \) is \( i^{th} \) trail solution vector.

Step 3: Find the best trail solution, which minimizes the objective function.

\[ Q(c) = Q_{best} \]

where \( Q_{best} \) is the best trail solution.

Step 4: Reduce the range using

\[ R(c+1) = R(c) \cdot (1-\beta) \]

where \( \beta \) is the reduction factor, whose typical value is 0.05.

Step 5: Increment count \( c \). The algorithm proceeds to Step 2, unless the best solution does not change for a pre-specified interval of generations.

The main reason for the success of DS algorithm is that lies in its local search ability. Since the values for the variables are always chosen around the best point determined in the previous iteration, there is a more likelihood of convergence to the optimum solution.

5. Methodology:

The pseudo code of the PSODS method,

Step 1: Get the data for the system
Step 2: Initialize randomly the searching points and velocities of the agents of PSO.
Step 3: Do
Step 4: Evaluate the objective function and update the inertia weight
Step 5: Modify the searching points and velocities
Step 6: If solution improves, then
Step 7: do fine tune the search region using
   Direct search method.
Step 8: end
Step 9: While (Termination criterion not met).

6. Numerical Results for the EDP:

Test cases of EDPs were studied in this section to illustrate the performance of the PSODS approach. The software was written in MATLAB6.1 and executed on a Pentium II 500 MHz personal computer. Due to the physical operation limitations of power plant components, the generating units exhibit a greater variation in the fuel cost functions. Industrial based algorithms like Lagrangian Multiplier (LM) method may cease to be applicable due to the non-smooth, non-differentiable fuel cost functions [2]. In this paper two different EDPs (based on fuel cost functions) were considered so as to validate the flexibility and effectiveness of the PSODS approach.

The following are the appropriate values of the various parameters used in the PSODS method to solve
the EDPs adopted in this paper. These values are typical and most fit for all the test cases.

\[ M=200, \quad w_{\max} =0.9, \quad w_{\min} =0.6, \quad c_1 = c_2 =2.0. \]

Termination criterion will stop the process early if the best solution has not improved for a significant number of iterations.

Case 1
This test case is composed of 13 generating units with the fuel cost functions take the valve-point loading effects into account. To simulate the valve-point loading effects of generating units, a recurring sinusoid component is added with the quadratic cost function. The system data can be found in [6]. The inclusion of valve point loading effects makes the modeling of the cost function of the generators more practical. This increases the non-linearity as well as number of local optima in the solution space. Also the solution procedure can easily trap in the local optima in the vicinity of optimal value. The fuel cost functions of the generating units are represented as follows,

\[ F_i(P_i) = a_i P_i^2 + b_i P_i + c_i + e_i \sin(f_i(P_{i_{\min}} - P_i)) \]  \hspace{1cm} (18)

where, \( e_i \) and \( f_i \) are cost coefficients of the \( i^{th} \) generating unit.

Table 1
Comparison of the results obtained using GA, SA, GA-SA and PSODS techniques for Case 1 for a load demand of 2520MW.

<table>
<thead>
<tr>
<th>Generator</th>
<th>Unit generation (MW)</th>
<th>GA</th>
<th>SA</th>
<th>GA-SA</th>
<th>PSODS</th>
</tr>
</thead>
<tbody>
<tr>
<td>z1</td>
<td>628.32</td>
<td>668.40</td>
<td>628.23</td>
<td>628.3094</td>
<td></td>
</tr>
<tr>
<td>z2</td>
<td>356.49</td>
<td>359.78</td>
<td>299.22</td>
<td>298.9996</td>
<td></td>
</tr>
<tr>
<td>z3</td>
<td>359.43</td>
<td>358.20</td>
<td>299.17</td>
<td>298.8181</td>
<td></td>
</tr>
<tr>
<td>a1</td>
<td>159.73</td>
<td>104.28</td>
<td>159.12</td>
<td>159.7441</td>
<td></td>
</tr>
<tr>
<td>a2</td>
<td>109.86</td>
<td>60.36</td>
<td>159.95</td>
<td>159.5509</td>
<td></td>
</tr>
<tr>
<td>a3</td>
<td>159.73</td>
<td>110.64</td>
<td>158.85</td>
<td>159.1718</td>
<td></td>
</tr>
<tr>
<td>a4</td>
<td>159.63</td>
<td>162.12</td>
<td>157.26</td>
<td>159.5712</td>
<td></td>
</tr>
<tr>
<td>b1</td>
<td>159.73</td>
<td>163.03</td>
<td>159.93</td>
<td>159.5940</td>
<td></td>
</tr>
<tr>
<td>b2</td>
<td>159.73</td>
<td>161.52</td>
<td>159.86</td>
<td>159.4003</td>
<td></td>
</tr>
<tr>
<td>b3</td>
<td>77.31</td>
<td>117.09</td>
<td>110.78</td>
<td>113.6156</td>
<td></td>
</tr>
<tr>
<td>c1</td>
<td>75.00</td>
<td>75.00</td>
<td>75.00</td>
<td>113.2250</td>
<td></td>
</tr>
<tr>
<td>c2</td>
<td>60.00</td>
<td>60.00</td>
<td>60.00</td>
<td>55.0000</td>
<td></td>
</tr>
<tr>
<td>c3</td>
<td>55.00</td>
<td>119.58</td>
<td>92.62</td>
<td>55.0000</td>
<td></td>
</tr>
</tbody>
</table>

Total Cost ($/hr) | 24398.23 | 24970.91 | 24275.71 | 24182.55

The total power demand of 2520 MW has to be generated. The same test case was solved using several heuristic techniques [6]. The results obtained using the PSODS method is compared with the results obtained using GA, SA and GA-SA [6]. The generations obtained using the PSODS and result comparisons are shown in Table 2.

Case 2
This test case comprises of three sub-systems and ten generating units with multiple-fuel options. Unlike conventional EDP which has a quadratic cost function, this system has piece-wise quadratic fuel cost function as multiple fuels are used for generation. The fuel cost functions of the generating units are represented as follows,

\[ F_i(P_i) = \begin{cases} a_{i1} + b_{i1} P_i + c_{i1} P_i^2, & P_{i_{\min}} \leq P_i \leq P_{i1} \\ a_{i2} + b_{i2} P_i + c_{i2} P_i^2, & P_{i1} \leq P_i \leq P_{i2} \\ a_{i3} + b_{i3} P_i + c_{i3} P_i^2, & P_{i2} \leq P_i \leq P_{i_{\max}} \end{cases} \]  \hspace{1cm} (20)

where, \( a_i, b_i, c_i \) are cost coefficients of the \( i^{th} \) generating unit for fuel type 1.

where, \( a_i, b_i, c_i \) are cost coefficients of the \( i^{th} \) generating unit for fuel type 2.

where, \( a_i, b_i, c_i \) are cost coefficients of the \( i^{th} \) generating unit for fuel type 3.

Fig 1. Convergence characteristics of PSODS for case 1

Fig 2. Convergence characteristics of the PSODS for Case 2 for \( P_{i3}=2700\text{MW} \)
The data for this system can be found in [10]. The generating units are expected to supply a load demand of 2400 MW, 2500 MW, 2600 MW and 2700 MW. The results of the PSODS method is compared with the results obtained using the Numerical method (NM), Enhanced Lagrangian Neural Network method (ELNN) [10], GA method [11] and EP method [12]. The results are shown in Table 2. The convergence characteristic is shown in Fig 2 for a load demand of 2700 MW.

In all the two test cases the performance of the PSODS was not affected by the generators fuel cost function model. Except for the third case all other cases are mentioned for single power demand. The average execution time of the PSODS method is 28.77 Sec for Case 1 and 25.33 Sec for Case 2 (for single power demand) compared to other algorithm reported in the literature for the test case problems adopted in this paper.

Some interesting observations can be made regarding the performance of the PSODS in terms of the reliability of producing quality solutions, searching efficiency as iteration proceeds, accuracy of the final solution and convergence characteristics when the number of agents are varied. These observations are summarized only for case 1 owing to the limited space, although the observations hold good for case 2 also.

The reliability of the PSODS method is illustrated in Fig 3 by executing the method for 100 trial runs with different random initial solutions. The consistency of the PSODS method in producing quality solution can be visualized from the figure. The solution procedure converges to an optimum point starting from any point in the exploring space. Since the agents movement is totally probabilistic, about 77% of the 100 trial runs produce quality solution thereby the PSODS method proves itself robust and superior. Also the convergence characteristic resembles the same for all the 100 trial runs. The searching efficiency of the PSODS is summarized in Table 7. It shows how the search progresses towards the optimum solution. The table shows the average, best and worst solutions generated by the PSODS and PSO after the 1, 15, 25, 35 iterations in all the 100 trial runs. It is clear from Table 7 that the search of the final solution has converged near the 35th iteration.

A typical convergence characteristic of the PSO for case 1 with and without the stretching function is shown in Fig 4. As seen the PSO stagnated after the 15th iteration and generated a solution which is local optimum. Whereas PSODS further explores to find much optimum solution than the one generated by the PSO. Also the PSODS is applied to solve the case 2 problem with various agents number. Result shows that, the number of agents above 200 does not have considerable influence on the convergence characteristics and quality of solution.

<table>
<thead>
<tr>
<th>P0 (MW)</th>
<th>NM</th>
<th>ELNN</th>
<th>GA</th>
<th>EP</th>
<th>PSODS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2400</td>
<td>488.50</td>
<td>481.74</td>
<td>482.00</td>
<td>481.79</td>
<td>481.72</td>
</tr>
<tr>
<td>2500</td>
<td>526.70</td>
<td>526.27</td>
<td>526.24</td>
<td>526.24</td>
<td>526.24</td>
</tr>
<tr>
<td>2600</td>
<td>574.03</td>
<td>574.41</td>
<td>574.40</td>
<td>574.39</td>
<td>574.38</td>
</tr>
<tr>
<td>2700</td>
<td>625.18</td>
<td>623.88</td>
<td>623.81</td>
<td>623.81</td>
<td>623.81</td>
</tr>
</tbody>
</table>

Whereas the number of agents below 50 especially 20 and less show a poor converging characteristics and finds it difficult to reach the optimum point. Also the PSODS does not guarantee a 100% success rate, in generating quality solution. For agents number above 200 the PSODS takes 30 -37 iterations to converge. A compromise size of the agents adopted in this paper is 200 for all the test cases as the number of function evaluations directly depends on the agents size and total number of iterations.

![Fig 3 Final fuel cost values for 100 trial runs of PSODS method](image)

<table>
<thead>
<tr>
<th>Iteration Number</th>
<th>Method</th>
<th>Fuel Cost ($/hr)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average</td>
<td>Best</td>
</tr>
<tr>
<td>1</td>
<td>PSO</td>
<td>24830.73</td>
</tr>
<tr>
<td></td>
<td>PSODS</td>
<td>24829.97</td>
</tr>
<tr>
<td>15</td>
<td>PSO</td>
<td>24339.46</td>
</tr>
<tr>
<td></td>
<td>PSODS</td>
<td>24256.29</td>
</tr>
<tr>
<td>25</td>
<td>PSODS</td>
<td>24336.23</td>
</tr>
<tr>
<td></td>
<td>PSODS</td>
<td>24207.92</td>
</tr>
<tr>
<td>35</td>
<td>PSODS</td>
<td>24336.23</td>
</tr>
<tr>
<td></td>
<td>PSODS</td>
<td>24185.33</td>
</tr>
</tbody>
</table>
7. Conclusion

An approach based on hybrid PSO-DS for solving the EDP is presented. The PSODS method is capable of dealing directly with load demand at various intervals of time in the scheduled horizon with no restrictions on the shape of the input-output cost function of the generating unit. The PSO is very fast compared to other evolutionary techniques, but it did not possess the ability to improve upon the quality of the solutions as the number of generations increased. When the solution of the PSO improves in a run, the region will be fine tuned with the DS method. The PSO explores the search space quickly and fine tunes the final solution. The performance of the PSODS method was tested for two EDP test cases and compared with the results reported in recent literature. The results show that the convergence property was not affected based on the shape of the fuel cost function. The advantage of the PSODS method is its ability in finding high quality solutions reliably with fast converging characteristics. The method can also be extended to solve the DEDP with more inequality constraints included such as prohibited operating zones, spinning reserves etc., and thereby more accurate dispatch results can be obtained for practical problems.

Acknowledgment

The first author would like to specially thank the management of Karunya Institute of Technology, Coimbatore, India.

References: