Available online at www.sciencedirect.com
SCIENCE DIRECT®
Electric Power Systems Research 71 (2004) 51–59
ELECTRIC POWER SYSTEMS RESEARCH
www.elsevier.com/locate/epsr

Hybrid PSO–SQP for economic dispatch with valve-point effect

T. Aruldoss Albert Victoire [a,*], A. Ebenezer Jeyakumar [b]

[a] *Department of Electrical and Electronics Engineering, Karunya Institute of Technology, Coimbatore 641 114, India*
[b] *Department of Electrical and Electronics Engineering, Anna University, GCT Campus, Coimbatore 641 013, India*

Received 9 July 2003; received in revised form 31 October 2003; accepted 9 December 2003

Abstract

This paper presents a novel and efficient method for solving the economic dispatch problem (EDP), by integrating the particle swarm optimization (PSO) technique with the sequential quadratic programming (SQP) technique. PSO is the main optimizer and the SQP is used to fine tune for every improvement in the solution of the PSO run. PSO is a derivative free optimization technique which produces results quickly and proves itself fit for solving large-scale complex EDP without considering the nature of the incremental fuel cost function it minimizes. SQP is a nonlinear programming method which starts from a single searching point and finds a solution using the gradient information. The effectiveness of the proposed method is validated by carrying out extensive tests on three different EDP with incremental fuel cost function takes into account the valve-point loadings effects. The proposed method out-performs and provides quality solutions compared to other existing techniques for EDP considering valve-point effects are shown in general.

Keywords: Particle swarm optimization; Sequential quadratic programming; Economic dispatch

1. Introduction

The primary objective of the economic dispatch problem (EDP) of electric power generation is to schedule the committed generating unit outputs so as to meet the required load demand at minimum operating cost while satisfying all unit and system equality and inequality constraints [1]. This makes the EDP a large-scale highly nonlinear constrained optimization problem. Improvements in scheduling the unit outputs can lead to significant cost savings. In traditional EDP, the cost function for each generator has been approximately represented by a single quadratic function and is solved using mathematical programming based on optimization techniques such as λ iteration method, gradient method, dynamic programming (DP) method, and so on [2]. These techniques require incremental fuel cost curves should be monotonically increasing to find global optimal solution. Whereas, the input output characteristics of large units are inherently highly nonlinear because of valve-point loadings, generating unit ramp rate limits, etc., and furthermore they may generate multiple local minimum points in the cost function.

For generating units, which actually having non-monotonically incremental fuel cost curves, the conventional method ignores or flattens out the portions of the incremental fuel cost curve that are not continuous or monotonically increasing. Hence, inaccurate dispatch result is induced. To obtain accurate dispatch results, approaches without restriction on the shape of incremental fuel cost functions are needed. Classical calculus-based techniques fail to address these types of problems satisfactorily. Unlike some traditional algorithms, dynamic programming [1] imposes no restrictions on the nature of the cost curves and therefore it can solve ELD problems with inherently nonlinear and discontinuous cost curves. This method, however, suffers from the “curse of dimensionality” or local optimality.

In this respect, stochastic search algorithms such as genetic algorithms (GAs) [3], evolutionary programming (EP) [1,4] and simulated annealing (SA) [5], may prove to be very effective in solving nonlinear ELD problems without any restrictions on the shape of the cost curves. Although these heuristic methods do not always guarantee discovering the globally optimal solution in finite time, they often provide a fast and reasonable solution (sub-optimal near

* Corresponding author. Fax: +91-422-2615615.
E-mail address: aruldoss@karunya.ac.in (T.A.A. Victoire).

0378-7796/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.epsr.2003.12.017

globally optimal). The SA method is a powerful optimization technique and it has the ability to find near global optimum solutions for the optimization problem. SA is applied in many power system problems. However, appropriate setting of the control parameters of the SA based algorithm is a difficult task and the speed of the algorithm is slow when applied to a real power system [6].

The evolutionary algorithms, EAs, (GA and EP) are search algorithms based on the simulated evolutionary process of natural selection, variation, and genetics. The evolutionary algorithms are more flexible and robust than conventional calculus-based methods. Both GA and EP can provide a near global solution [1]. However, the encoding and decoding schemes essential in the GA approach makes it to take longer time for convergence. EP differs from traditional GAs in two aspects: EP uses the control parameters (real values), but not their codings as in traditional GAs, and EP relies primarily on mutation and selection, but not crossover, as in traditional GAs. Hence, considerable computation time may be saved in EP. Although GA and EP seem to be good methods to solve optimization problems, when applied to problems consists of more number of local minima the solutions obtained from both methods are just near global optimum ones. And also GA and EP take long computation times in order to obtain the solutions for such problems [7]. Therefore, hybrid methods combining two or more optimization methods were introduced [7–9].

Particle swarm optimization (PSO) [10] is one of the modern heuristic algorithms under the EAs and gained lots of attention in various power system applications [10–13]. PSO can be applied to nonlinear and non-continuous optimization problems with continuous variables. It has been developed through simulation of simplified social models. PSO is similar to the other evolutionary algorithms in that the system is initialized with a population of random solutions. However, each potential solution is also assigned a randomized velocity, and the potential solutions, call agents, corresponding to individuals. Each agent in PSO flies in the n-dimensional problem space with a velocity which is dynamically adjusted according to the flying experiences of its own and its colleagues.

Generally, the PSO is characterized as a simple heuristic of well balanced mechanism with flexibility to enhance and adapt to both global and local exploration abilities [13]. It is a stochastic search technique with reduced memory requirement, computationally effective and easier to implement compared to other EAs. PSO developed by Dr. Kennedy and Dr. Eberhart, shares some of the common features available in other EAs, except the selection procedure. Also, PSO will not follow survival of the fittest, the principle of other EAs. PSO when compared to EP has very fast converging characteristics; however it has a slow fine tuning ability of the solution. Also PSO has a more global searching ability at the beginning of the run and a local search near the end of the run. Therefore, while solving problems with more local optima, there are more possibilities for the PSO to explore local optima at the end of the run.

To overcome this drawback a hybrid method that integrates the PSO with a gradient search algorithm called SQP [14] is proposed in this paper. In the beginning of the run PSO has more possibilities to explore a large space and therefore the agents are freer to move and sit on various valleys. The best value of all the agents will be taken as the initial starting point for the SQP and will be fine tuned. Thus, the possibility of exploring a global minimum in problems with more local optima is increased. The search will continue until a termination criterion is satisfied. To validate the performance of the proposed approach three economic dispatch problems with incremental fuel cost functions taking into account the valve-point loading effects were tested and the results obtained were compared with those obtained using PSO, hybrid EP–SQP technique and other techniques reported in recent literatures [1,6].

2. EDP formulation

The classic EDP minimizes the following incremental fuel cost function associated to dispatchable units [16]:

$$F_{\mathrm{T}} = \sum_{i=1}^{N} F_i(P_i) \tag{1}$$

The inclusion of valve-point loading effects makes the modeling of the incremental fuel 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 incremental fuel cost function of the generating units with valve-point loadings are represented as follows [3]:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i + |e_i \,\mathrm{Sin}(f_i(P_{i\,\min} - P_i)| \tag{2}$$

where, a_i, b_i, c_i are cost coefficients and e_i, f_i are constants from the valve-point effect of the ith generating unit subject to the following equality and inequality constraints:

(a) real power balance:

$$\sum_{i=1}^{N} P_i = P_{\mathrm{D}} + P_{\mathrm{Loss}} \tag{3}$$

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

$$P_{\mathrm{Loss}} = \sum_{m=1}^{N}\sum_{n=1}^{N} P_m B_{mn} P_n \tag{4}$$

(b) real power generation limit:

$$P_{i\,\min} \le P_i \le P_{i\,\max} \tag{5}$$

where, F_T, total production cost ($/h); $F_i(P_i)$, incremental fuel cost function ($/h); P_i, real power output of the ith 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 ith unit (MW); $P_{i\,max}$, maximum limit of the real power of the ith 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 incremental fuel cost function can be modeled in a more practical fashion by including the valve-point effects [17,18]. The generating units with multi-valve steam turbines exhibit a greater variation in the fuel cost functions. The valve-point effects introduce ripples in the heat-rate curves, thereby the number of local optima is increased. Hence, a technique that overcomes these complexities has to be evolved.

3. Particle swarm optimization [10]

Understanding the emergence and evolution of biological and social order has been a fundamental goal of evolutionary theory. Here, we discuss another type of biological system–social system, more specifically, the collective behaviors of simple individuals interacting with their environment and each other. PSO is one of the modern heuristic algorithms developed by Kennedy and Eberhart [10]. It has been developed through simulation of simplified social models. Compared to other evolutionary techniques, the advantages of PSO are that it 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:

$$v_j^{t+1} = wv_j^t + c_1\mathrm{rand}(P_{\mathrm{best}\,j} - s_j^t) + c_2\mathrm{rand}(G_{\mathrm{best}} - s_j^t) \quad (6)$$

where, 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; $P_{\mathrm{best}\,j}$, P best of agent j; G_{best}, G best of the group; w, inertia weight.

The inertia weight is usually calculated using the following expression:

$$w = w_{max} - (w_{max} - w_{min}) \times \mathrm{iter}/\mathrm{iter}_{max} \quad (7)$$

where, w_{max}, initial weight; w_{min}, final weight; iter_{max}, maximum iteration number; iter, current iteration number.

Searching point of each agent is modified according to the following equation:

$$s_j^{t+1} = s_j^t + v_j^{t+1} \quad (8)$$

Improved performance of the PSO can be obtained by carefully selecting the inertia weight w, c_1, and c_2. For larger values of inertia weight PSO has global exploration feature and vice-versa [15]. Even though there need a trade-off between the quality of solution and fine tuning of the PSO while selecting the parameters.

3.1. Phase I: PSO based EDP

Let $p_j = [P_1, P_2, \ldots, P_r, \ldots, P_N]$ be the initial searching point of the jth agent in the space to be evolved. Similarly for all the M agents the searching points are initialized. The elements of the p_j vector are the real power outputs of the committed generating units, which are subjected to the unit's generation limit constraints, required power demand constraint and other constraints imposed on the problem. To meet exactly the required power demand a dependent generating unit is arbitrarily selected from among the committed N units. Let P_r be the real power generated by the dependent generating unit and is calculated using Eq. (3) given by:

$$P_r = P_D + P_{Loss} - \sum_{\substack{i=1\\ i\neq r}}^{N} P_i$$

(a) Generation of initial conditions of each agent:

The initial conditions of each agent have to be generated randomly within the limits specified for each agent. In the EDP the initial searching points are the initial random real power outputs of the generators denoted by p_j, $j = 1, 2, \ldots, M$, is determined by setting its ith components $P_i \sim U(P_{i\,min}, P_{i\,max})$, where $i = 1, 2, \ldots, N$. Where $U(P_{i\,min}, P_{i\,max})$ denotes a uniform distribution of variables ranging over the interval $[P_{i\,min}, P_{i\,max}]$.

(b) Evaluation of each agent:

Each agent is evaluated using the fitness function of the economic dispatch problem to minimize the incremental fuel cost function given by (1),

$$O_j(t) = \mathrm{Min}\sum_{i=1}^{N} F_i(P_i) \quad (9)$$

Search for the best value of all the fitness function values $O_{j,\,\mathrm{best}}(t)$ from $O_j(t)$, $j = 1, 2, \ldots, M$ evaluated using the M agents. Set the agent associated with $O_{j,\,\mathrm{best}}(t)$ as the global best p_{Gbest} of all the agents. The best fitness value of each agent upto the current iteration is set to as local best of that agent $P_{\mathrm{Lbest},\,j}$.

(c) Modification of each searching point:

Using the global best and the local best of each agent upto the current iteration, the searching point of each agent has to be modified according to the following expression,

$$p_j(t) = v_j(t) + p_j(t-1) \tag{10}$$

where,

$$v_j(t) = w(t)v_j(t-1) + c_1 \text{rand}_1(p_{\text{Lbest},j} - p_j(t-1)) + c_2 \text{rand}_2(p_{\text{Gbest}} - p_j(t-1)) \tag{11}$$

Thus, the new searching points were explored for the next iteration to further exploit the search. Once the new searching points were determined the inertia weight has to be modified.

(d) Modification of the global and the local bests:

Each agent should be evaluated using the fitness function of the economic dispatch as was done in point (b). The p_{Gbest} and $p_{\text{Lbest},\,j}$ have to be modified according to the present fitness function values evaluated using the new search points of the agents. If the best fitness value of all the fitness function values is better than the $O_{j\,\text{best}}$ then change $p_{\text{Gbest}}(t-1)$ to this value of the searching point of the corresponding agent contribute for this best fitness value. Similarly the local best of other agents in the population should be changed accordingly if the present fitness function value is better than the previous.

(e) Termination criteria:

Repeat the procedure from (c) until the maximum number of iterations reached or no improvement in the solution for a specified maximum number of iterations.

From the above discussion about the solution of the EDP using the PSO approach it is clear that the method solves the problem without considering the shape of the incremental fuel cost functions of the generating units.

4. Sequential quadratic programming (SQP) [14]

SQP method seems to be the best nonlinear programming methods for constrained optimization. It outperforms every other nonlinear programming method in terms of efficiency, accuracy, and percentage of successful solutions, over a large number of test problems. The method resembles closely to Newton's method for constrained optimization just as is done for unconstrained optimization. At each iteration an approximation is made of the Hessian of the Lagrangian function using a BFGS quasi-Newton updating method. This is then used to generate a quadratic programming (QP) sub-problem whose solution is used to form a search direction for a line search procedure.

4.1. Phase II: SQP based EDP

First let us formulate the QP sub-problem for the problem as stated by (1) subject to (3) and (5).

$$\text{Min}\ \nabla F_{\text{T}}(P_k)^{\text{T}} d_k + \tfrac{1}{2} d_k^{\text{T}} H_k d_k \tag{12}$$

subject to

$$c(P_k) + \nabla c(P_k)^{\text{T}} d_k = 0 \tag{13}$$

$$P_{\min} \le P_k + d_k \le P_{\max} \tag{14}$$

where, H_k is the Hessian matrix of the Lagrangian function at the kth iteration; d_k, search direction at the kth iteration; P_k, real power vector at the kth iteration; and $c(P_k)$ is the constraint given by Eq. (3)

$$L(P, \lambda) = F_{\text{T}}(P) + c(P)^{\text{T}}\lambda \tag{15}$$

and is constructed from a quasi-Newton update formula given by:

$$H_{k+1} = H_k + \frac{q_k q_k^{\text{T}}}{q_k^{\text{T}} s_k} - \frac{H_k^{\text{T}} s_k^{\text{T}} s_k H_k}{s_k^{\text{T}} H_k s_k} \tag{16}$$

where

$$s_k = P_{k+1} - P_k \tag{17}$$

$$q_k = \nabla L(P_{k+1}, \lambda_{k+1}) - \nabla L(P_k, \lambda_{k+1}) \tag{18}$$

For each iteration of the QP sub-problem the direction d_k is calculated using the above Eq. (12). The solution obtained is used to form a new iterate given by:

$$P_{k+1} = P_k + \alpha_k d_k \tag{19}$$

The step length value α_k is determined to produce a considerable reduction in an augmented Lagrangian merit function (20):

$$L_{\text{A}}(P, \lambda, \rho) = F_T(P) - \lambda^{\text{T}}(P) + \frac{\rho}{2} c(P)c(P) \tag{20}$$

where, λ is the vector of Lagrangian multiplier; ρ is a non-negative scalar

The procedure will be repeated until the value of s_k has reached some tolerance value.

5. Solution methodology

The pseudo code of the proposed solution methodology that integrates the PSO with SQP for EDP with valve-point effects can be summarized as follows:

Step 1: get the data for the system.
Step 2: initialize randomly the searching points, velocities of the agents of PSO and count t.
Step 3: do.
Step 4: evaluate the objective function and update the inertia weight and count t.
Step 5: identify the $G_{\text{best}}(t)$ of the current run t.
Step 6: is $G_{\text{best}}(\text{t}) < G_{\text{best}}(\text{t}-1)$.
Step 7: solve the EDP using the SQP method with the current $G_{\text{best}}(t)$ of the PSO as the starting point.

Step 8: replace $G_{\text{best}}(t)$ with the final solution obtained using the SQP.
Step 9: otherwise goto Step 10.
Step 10: modify the velocities and searching points.
Step 11: while (termination criterion not met).

The termination is done when there is no improvement in the solution for a specified number of iterations.

6. Simulation results

The proposed PSO–SQP approach was tested with three test cases of EDP with valve-point effects. To simulate the valve-point loading effects of generating units, a recurring sinusoid component is added with the quadratic cost function. The software was written in MATLAB 6.1 and executed on a Pentium II 500 MHz personal computer. Hereinafter, the results represent the average of 30 runs of the proposed method for all the three test cases. In the following section, optimal range of inertia weight and values of the weighting factors of the PSO–SQP technique to solve the three test cases will be discussed.

6.1. Optimal estimation of PSO parameters

Selection of the inertia weight w and weighting factors c_1 and c_2 considerably affects the performance of the PSO. The inertia weight controls the local and global exploration capabilities of the PSO [10]. Large inertia weight enables the PSO to explore globally and small inertia weight enables it to explore locally. In the proposed hybrid PSO–SQP technique, PSO is used to explore globally the search space, since the local exploration is done using the SQP. Therefore, to fix an optimal range of inertia weight, to solve the three test cases considered in this article, experiments were conducted using the proposed hybrid PSO–SQP by varying the inertia weights from 1.2 to 0.4, in steps of 0.01. For all the three test cases, the agents size and maximum number of iteration are fixed at 100 and 100, respectively. Thirty independent trial runs were conducted to bring out the optimal range of inertia weight for the proposed hybrid PSO–SQP technique.

Fig. 1 shows the plot of the relative frequency of convergence towards quality solution (better solution, compared to the solutions produced by any other technique reported) versus the inertia weight for all the three test cases. It is clear from the Fig. 1 that, for test case 1, for all the inertia weights the hybrid PSO–SQP technique has produced quality solution. For test case 2, for all the inertia weights above 0.49 the technique produced quality solution. And for test case 3, for all the inertia weights above 0.63 the technique produced quality solution. To ensure reliability in producing quality solutions by the proposed technique the relative frequency of convergence towards quality solution is fixed as above 70%. Thus, the hybrid SO–SQP technique has reliably produced the quality solutions for inertia weights above 0.57, 0.69, and 0.99 for test cases 1, 2, and 3, respectively.

Fig. 2, shows the average computation time taken by the hybrid PSO–SQP technique to solve the three test cases for various inertia weights. It is clear from the Fig. 2 that, when the technique produces quality solutions for lower values of inertia weights it produces it quickly, and it takes more time when values of inertia weights are large. Therefore, there need to be a compromise between the time taken and reliability in producing the quality solutions. With regard to ensuring quality solutions at a reasonable computation time for wide ranges of problems the optimal inertia weight range of 0.99–0.6 has been chosen. This inertia weight range is found to be general and effective in solving the problems considered in this article.

The optimal values for the weighting factors c_1, c_2 are chosen by conducting experiments in all the three test cases considered in this article. Weighting factors c_1, c_2 are assigned values ranging from 1 to 5 in steps of 1. Three different cases have been studied. First case considers always c_1 greater than c_2. In this case the technique takes more time

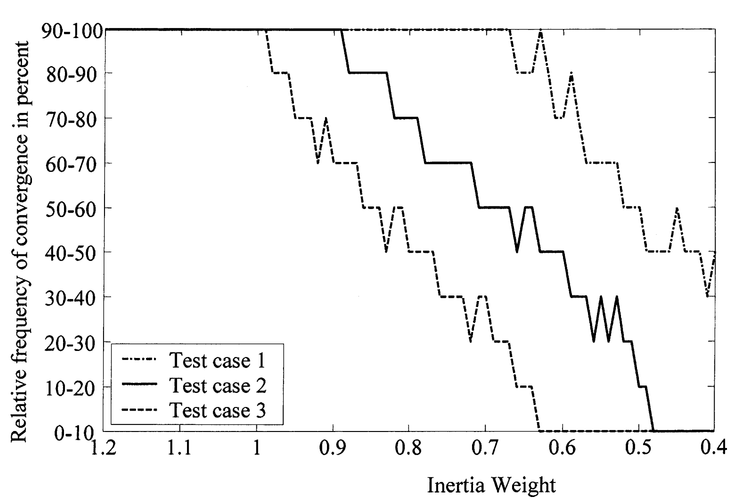

Fig. 1. Relative frequency of convergence in percent using hybrid PSO–SQP.

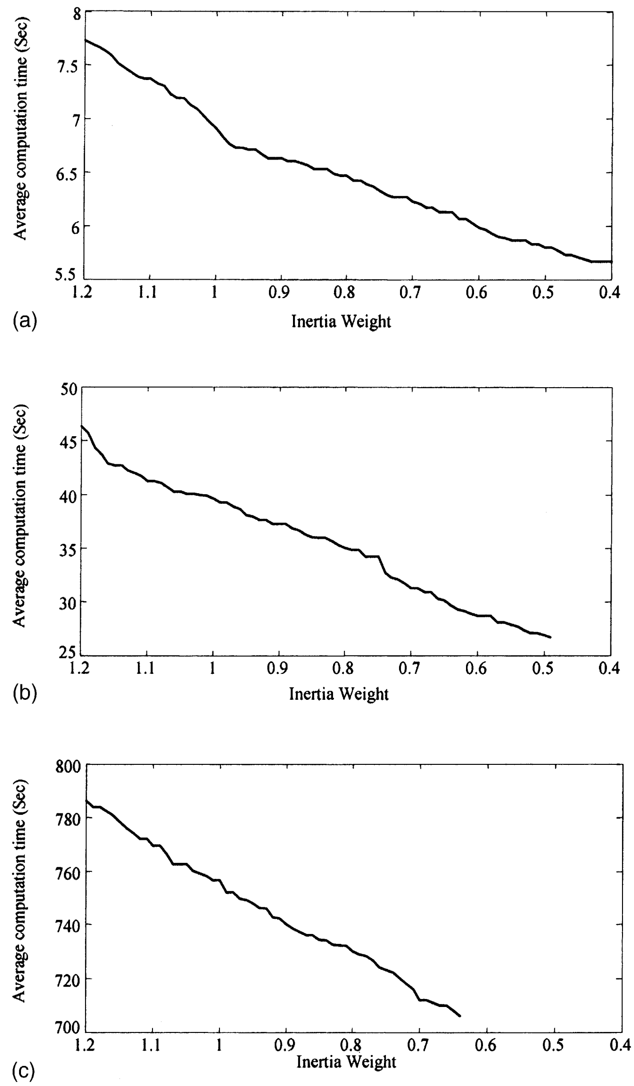

Fig. 2. Average computation time for various inertia weights for three test cases using hybrid PSO–SQP. (a) Test case 1; (b) test case 2; and (c) test case 3.

to converge. Second case considers always c_1 less than c_2. In this case the technique takes less time to converge, but the solution obtained is not a quality solution. Third case considers both c_1 and c_2 equal. Results obtained in this case seem to be better compared to the other two cases. The technique could find the solution quickly as well as effectively. Even in this case, the hybrid PSO–SQP technique is found to be general and effective when $c_1 = c_2 = 2$. This value for the constriction factors is also found to be suitable for a wide range of power system problems [10].

6.2. Numerical solutions

To compare the results obtained using the hybrid PSO–SQP and the hybrid EP–SQP [7] techniques, the coding of the hybrid EP–SQP technique was also written in MATLAB 6.1. The simulation parameters of the hybrid PSO–SQP technique for all the test systems are fixed as follows, $w_{max} = 0.99$, $w_{min} = 0.6$, $c_1 = c_2 = 2$, and iter = 30 for test case 1 and 100 for test case 2 and 3. Similarly, for the hybrid EP–SQP technique all simulation parameters for all the test systems are same as in [7] except the number of candidates = 30 for test case 1 and 100 for test case 2 and 3, and $iter_{max} = 30$ for test case 1 and 100 for test case 2 and 3. Also to validate the comparison of results obtained using the proposed technique with the results obtained using the EP, PSO, and EP–SQP techniques, solution procedure of the hybrid PSO–SQP technique is terminated when the maximum number of iterations is reached.

6.2.1. Case 1

This test case comprises of three generating units. The expected power demand to be met by all the three generating units is 850 MW. The system data can be found from [3]. The size of the agents are varied from 10 to 50 insteps of 10 and experimented using the proposed method. To show the effectiveness of integrating the SQP with the PSO, the problem was also experimented using the standard PSO [10] with inertia weight alone. The inertia weight for the standard PSO is varied from 0.9 to 0.4 as these values are accepted as typical for solving wide varieties of problems [10]. In the 30 trial runs the PSO has produced a best solution of \$8234.07 for 18 times. In most of the trial runs the agents could able to locate the global optimum region (not exactly the global optimum) in the beginning of the run itself. But when the run progresses the agents leave the space as they swarm in a region where they found it more optimum than they visited previously.

In a typical trial run, one of the 30 agents of the PSO had a fitness value of \$8245.7 in the third iteration with $w = 0.85$, but when the run progressed, the agent left to a new region with a fitness value of \$8241.6. Actually, the agent had left the region which is actually a region having the global optimum point. This happened due to the global exploration capability of the PSO during the beginning of the run. To overcome this, the SQP method is incorporated with the standard PSO, and fine tuned the global best agent position for every improvement in the solution. Thus, the same test case was solved using the standard PSO with the SQP incorporated. The PSO will be initiated to solve the problem, when the region of global optimum is reached by any one of the agents in the PSO, the region is fine tuned using SQP by accepting this as the starting point for the algorithm and the global optimum point is thus explored in the early iterations of the PSO run. A best solution of \$8234.07 was recorded for all the 30 trial runs by the proposed method within an average of five iterations of every run. The mean time for the simulation of the proposed method is 3.37 s compared to 4.37 s of the standard PSO.

The dispatch results using the proposed method, PSO, GA, EP, and EP–SQP are shown in Table 1. From the Table 1 it is clear that GA approach did not meet the load demand. Also it needs encoding and decoding schemes to feasibly run

Table 1
Dispatch results for a $P_D = 850$ MW for case 1

Method	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_D (MW)
GA	398.700	399.600	50.100	848.400
EP	300.264	400.000	149.736	850.000
EP–SQP	300.267	400.000	149.733	850.000
PSO	300.268	400.000	149.732	850.000
PSO–SQP	300.267	400.000	149.733	850.000

Table 2
Comparison of fuel costs for case 1

Method	Mean time (s)	Best cost ($/h)	Mean cost ($/h)
GA	35.8	8222.07	8234.72
EP	6.78	8234.07	8234.16
EP–SQP	5.12	8234.07	8234.09
PSO	4.37	8234.07	8234.72
PSO–SQP	3.37	8234.07	8234.07

the algorithm. Other four methods EP, EP–SQP, PSO and PSO–SQP did not need encoding or decoding the decision variables. The final fuel cost of all the five methods are summarized in Table 2. It is clear that the mean cost value and simulation time obtained by the proposed method is comparatively less compared to all the other methods.

The problem has a number of local optimum points as there are more possibilities for any method to stick on any one of the local optimum points. Fig. 3 shows the convergence characteristics of the PSO and PSO–SQP methods for case 1. It is clear in the beginning of the run itself the PSO explored near the global optimum point. As run progress it left the global optimum point and takes few more runs to converge towards the global optimum. Whereas, the PSO–SQP method discovered the global optimum within three runs of the PSO method, as the SQP fine tunes for every improvement of the solution in the PSO run.

6.2.2. Case 2

This test case comprises of thirteen generating units, The complexity and non-linearity to the solution procedure is increased. The required power demands to be met by all the thirteen generating units is 1800 [1] and 2520 MW [5]. The system data can be found from [5]. The final fuel costs obtained using the EP, EP–SQP, PSO, and PSO–SQP methods for power demand of 1800 MW were summarized in Table 3. Table 4 reports the dispatch results of the various methods [6], EP–SQP and the proposed method for a load demand of 2520 MW. The problem is solved for two different power demands inorder to show the effectiveness of the proposed method in producing quality solutions. The results depict the minimum fuel cost solution obtained for both power demands by the proposed method with a reduced simulation time. In the thirty trial runs with different initial random solutions, the proposed method has proven it as reliable solution procedure by producing 21 times the global optimum shown in Tables 3 and 4.

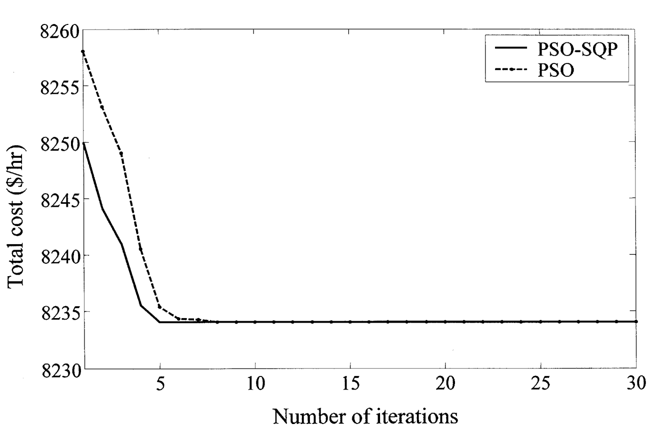

Fig. 3. Convergence characteristics of the proposed method and PSO for case 1 for a $P_D = 850$ MW.

Table 3
Comparison of fuel costs for case 2 for a $P_D = 1800$ MW

Method	Mean time (s)	Best cost ($/h)	Mean cost ($/h)
EP	157.43	17,994.07	18,127.06
EP–SQP	121.93	17,991.03	18,106.93
PSO	77.37	18,030.72	18,205.78
PSO–SQP	33.97	17,969.93	18,029.99

It is clear from the Tables 3 and 4, the mean cost value and simulation time obtained by the proposed method is comparatively less compared to all the other methods. Fig. 4 shows the convergence characteristics of the PSO–SQP method for case 2. The size of the agents is varied from 20 to 100 in steps of 20 and the results summarized are for a size of 100. The average number of iterations to reach the optimum solution is 27–30 in all the 30 runs.

6.2.3. Case 3

This test case comprises of 40 generating units. The number of local optima, complexity and non-linearity to the

Table 4
Dispatch results for a $P_D = 2520$ MW for case 2

Generator	Unit generation (MW)				
	GA	SA	GA–SA	EP–SQP	PSO–SQP
z1	628.32	668.40	628.23	628.3136	628.3205
z2	356.49	359.78	299.22	299.1715	299.0524
z3	359.43	358.20	299.17	299.0474	298.9681
a1	159.73	104.28	159.12	159.6399	159.4680
a2	109.86	60.36	159.95	159.6560	159.1429
a3	159.73	110.64	158.85	158.4831	159.2724
a4	159.63	162.12	157.26	159.6749	159.5371
b1	159.73	163.03	159.93	159.7265	158.8522
b2	159.73	161.52	159.86	159.6653	159.7845
b3	77.31	117.09	110.78	114.0334	110.9618
c1	75.00	75.00	75.00	75.0000	75.0000
c2	60.00	60.00	60.00	60.0000	60.0000
c3	55.00	119.58	92.62	87.5884	91.6401
Total cost ($/h)	24,398.23	24,970.91	24,275.71	24,266.44	24,261.05

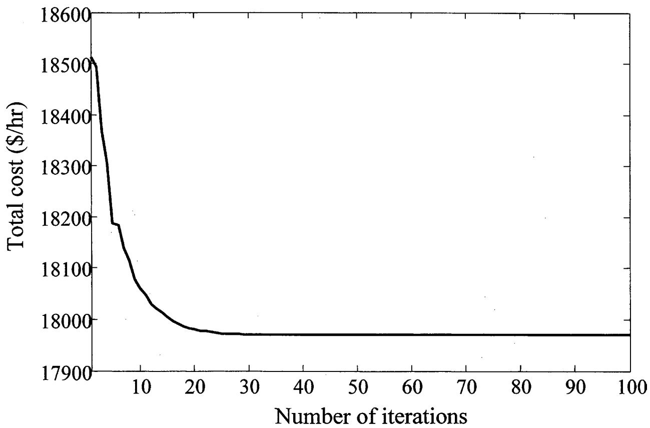

Fig. 4. Convergence characteristics of the proposed method for case 2 for a $P_D = 1800$ MW.

Table 5
Comparison of fuel costs for case 3

Method	Mean time (s)	Best cost ($/h)	Mean cost ($/h)
EP	1167.35	122,624.35	123,382.00
EP–SQP	997.73	122,323.97	122,379.63
PSO	933.39	123,930.45	124,154.49
PSO–SQP	733.97	122,094.67	122,245.25

solution procedure is enormously increased. The required power demand to be met by all the forty generating units is 10,500 MW. The system data can be found from [1]. The final fuel costs obtained using the EP, EP–SQP, PSO, and PSO–SQP methods were summarized in Table 5. It is clear from Table 5, the mean cost value and simulation time obtained by the proposed method is comparatively less compared to the EP, EP–SQP, and PSO methods. Fig. 5 shows the convergence characteristics of the PSO–SQP methods for case 3. The size of the agents is varied from 20 to 100 in steps of 20 and the results summarized are for a size of 100. The average number of iterations to reach the optimum solution is 36–40 in all the 30 trial runs.

7. Discussion

Traditionally to solve the EDP effectively, conventional techniques need the incremental fuel cost curves to be of featured monotonically increasing and continuous. But practically the generating units, which actually having non-monotonically incremental fuel cost curves when solved, the conventional method ignores or flattens out the portions of the incremental fuel cost curve that are non-continuous or non-monotonically increasing. To obtain accurate dispatch results, technique without restriction on the shape of incremental fuel cost functions has to be modeled.

PSO is a recent tool for solving complex optimization problems, being attracted by the researchers in various diverse fields. It was also effectively used in solving complex problems in the power system field. It is faster in finding quality solution compared to any evolutionary computation technique but finds it difficult while exploring complex functions. It leads to premature convergence and also has a poor fine tuning of the final solution.

To overcome these drawbacks, PSO was integrated with SQP. This new technique is used to solve the EDP with incremental fuel cost functions take valve-point effects into account. Some of the comments on PSO method to solve EDP:

- the method does not depend on the nature of the function it minimizes. Thus, approximations made in conventional techniques are avoided;
- the technique is insensitive to the initial searching points thus ensuring a quality solution with high probability;
- the memory requirement is very low;
- the convergence is not affected by the inclusion of more constraints;
- the method is more robust and flexible when used in complicated and non-continuous search spaces.

Unlike the conventional economic dispatch algorithms, the PSO method can easily handle other operating

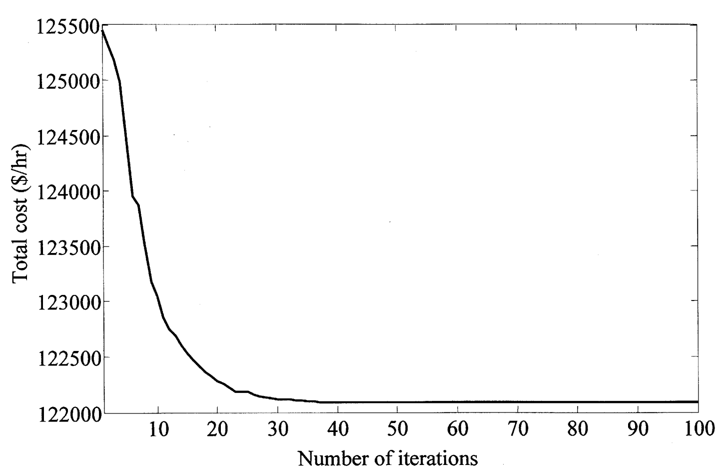

Fig. 5. Convergence characteristics of the proposed method for case 3 for a $P_D = 10{,}500$ MW.

constraints like spinning reserves, transmission lines capacity limits, etc.

SQP proves itself as a best non-linear programming method to solve constrained optimization problem. The method is sensitive to the initial point. It guarantees local optima as it follows a gradient search direction from the starting point towards the optimum point. The method when integrated with the PSO produces quality solutions as compared to the one produced by these techniques when applied separately.

8. Conclusion

An approach by integrating the PSO with the SQP for solving the EDP with valve-point effects is presented. PSO with inertia weight has the ability of global exploration at the beginning of the run and has a local exploration at the ending of the run. Thus, there are possibilities for it to miss the global optimum point while in the beginning of the run, and will fine tune the point at the end of the run which may not be the global optimum. To overcome this, SQP method is integrated and used to fine tune the improving solutions of the PSO run. Whenever, there is an improvement in the solution in the PSO run that will be taken as the starting point of the SQP and fine tuned. The PSO 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. It is very fast compared to other evolutionary techniques in exploring the search space. The SQP explores the search space quickly with a gradient direction and guarantees a local optimum solution. The performance of the PSO–SQP method was tested for three EDP test cases with valve-point effects included and compared with the results obtained using the EP–SQP technique and the results reported in recent literature. The results show that the convergence property was not affected based on the shape of the incremental fuel cost function. The advantage of the PSO–SQP method is its ability in finding high quality solutions reliably with fast converging characteristics. The method can also be extended to solve the dynamic EDP with more inequality constraints included such as transmission limits, voltage limits, prohibited operating zones, spinning reserves, etc., and thereby more accurate dispatch results can be obtained for practical problems in a reasonably good computation time.

References

[1] N. Sinha, R. Chakrabarti, P.K. Chattopadhyay, Evolutionary programming techniques for economic load dispatch, IEEE Trans. Evol. Comput. 7 (1) (2003) 83–94.

[2] D. Srinivasan, F. Wen, C.S. Chang, A.C. Liew, A survey of evolutionary computing in power systems, IEEE Proc. (1996) 35–41.

[3] D.C. Walters, G.B. Sheble, Genetic algorithm solution of economic dispatch with valve-point loadings, IEEE Trans. Power Syst. 8 (3) (1993) 1325–1331.

[4] H.T. Yang, P.C. Yang, C.L. Huang, Evolutionary programming based economic dispatch for units with non-smooth incremental fuel cost functions, IEEE Trans. Power Syst. 11 (1) (1996) 112–118.

[5] K.P. Wong, Y.W. Wong, Genetic and genetic/simulated-annealing approaches to economic dispatch, IEEE Proc. Gener. Trans. Distrib. 141 (5) (1994) 507–513.

[6] D. Bhagwan Das, C. Patvardhan, Solution of Economic Load Dispatch using real coded Hybrid Stochastic Search, Int. J. Electr. Power Energy Syst. 21 (1999) 165–170.

[7] P. Attaviriyanupap, H. Kita, E. Tanaka, J. Hasegawa, A hybrid EP and SQP for dynamic economic dispatch with nonsmooth incremental fuel cost function, IEEE Trans. Power Syst. 17 (2) (2002) 411416.

[8] W.M. Lin, F.S. Cheng, M.T. Tsay, Nonconvex economic dispatch by integrated artificial intelligence, IEEE Trans. Power Syst. 16 (2) (2001) 307–311.

[9] S.C. Lee, Y.H. Kim, An enhanced Lagrangian neural network for the ELD problems with piecewise quadratic cost functions and nonlinear constraints, Electr. Power Syst. Res. 60 (2002) 167–177.

[10] J. Kennedy, R. Eberhart, Swarm Intelligence, Morgan Kaufmann Publishers, 2001.

[11] H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, Y. Nakanishi, A particle swarm optimization for reactive power and voltage control considering voltage security assessment, IEEE Trans. Power Syst. 15 (4) (2000) 1232–1239.

[12] M.A. Abido, Optimal design of power system stabilizers using particle swarm optimization, IEEE Trans. Energy Convers. 17 (3) (2002) 406–413.

[13] M.A. Abido, Optimal power flow using particle swarm optimization, Int. J. Electr. Power Energy Syst. 24 (7) (2002) 563–571.

[14] P.T. Boggs, J.W. Tolle, Sequential Quadratic Programming, Acta Numerica, Cambridge University Press, Cambridge, 4 (1995) 1–52.

[15] R. Eberhart, Y. Shi, comparing inertia weights and constriction factors in particle swarm optimization, in: Proceedings of the Congress on Evolutionary Computation (CEC2000), 2000, pp. 84–88.

[16] A.J. Wood, B.F. Wollenberg, Power Generation, Operation and Control, Wiley, New York, 1984.

[17] H.H. Happ, Optimal power dispatch—a comprehensive survey, IEEE Trans. PAS 96 (1977) 841–854.

[18] B.H. Chowdhury, S. Rahman, A review of recent advances in economic dispatch, IEEE Trans. Power Syst. 5 (4) (1990) 1248–1259.