Hybrid Differential Evolution and Particle Swarm Optimization Based Solutions to Short Term Hydro Thermal Scheduling

T JAYABARATHI, SANDEEP CHALASANI, ZAMEER AHMED SHAIK,
NISHCHAL DEEP KODALI
School of Electrical Sciences
Vellore Institute of Technology University
Vellore-632014, Tamil Nadu
INDIA
t_barathi63@yahoo.com, http://www.vit.ac.in

Abstract: In this paper, a hybrid differential evolution and a particle swarm optimization based algorithms are proposed for solving the problem of scheduling the hydro thermal generation for a short term. The efficient scheduling requires minimizing the operating cost of the thermal plants. A wide range of constraints both hydraulic and thermal like power balance constraint, thermal plant limits, hydro plant limits, reservoir volumes and discharge constraints are fully taken into consideration. The hydraulic continuity restrictions are also considered. The proposed HDE and PSO algorithms are implemented for a test system. .The program for these two algorithms has been developed in Matlab 6.5 platform. To validate the efficiency of the algorithms the test results are compared with those obtained by other conventional and non-conventional methods. It is shown that the proposed techniques yield optimal solutions when compared to other non-conventional methods.

Key-Words: - particle swarm optimization, Hybrid differential evolution, Hydro-Thermal scheduling,

Nomenclature

J	the interval in the schedule horizon 1,2,3,…, z
$f_j(PT_j)$	\$ h^{-1} fuel cost of the equivalent thermal generator operating at generation level.
PT_j	Thermal generation in jth interval
PH_j	hydro generation in jth interval
q_j	water discharge rate in jth interval
PD_j	total load demand in jth interval
PL_j	total electric loss between the hydro plant and the load in jth interval
n_j	number of hours in the jth interval.
R_j	water inflow rate into the storage reservoir in jth interval.
S_j	spillage discharge rate in jth interval.
V_j	volume of water stored in reservoir at the end of the jth interval.
k	a constant.
$g(.)$	Water discharge rate function.
V_{jmin}	Minimum amount of water that has to be in the reservoir.
V_{jmax}	Maximum limit of the reservoir volume
PT_{min}	Minimum limit of thermal generation
PT_{max}	Maximum limit of thermal generation
PH_{min}	Minimum limit of hydro generation
PH_{max}	Minimum limit of hydro generation

1 Introduction

The scheduling is a daily planning task. In view of the increase in demand for electricity for purposes such as industrial, agricultural, commercial and domestic together with the high cost of fuel as well as its limited reserve, considerable attention is being given to the hydrothermal coordination problem. Besides the insignificant incremental cost involved in the hydro generation, the problem of minimizing the operational cost of a thermal plant can be now reduced essentially to that of minimizing the fuel cost for thermal plants under the constraints of the water available for the hydro generation in a given period of time.

The primary objective of the short term hydro thermal scheduling is to find the generation levels of the hydro and thermal units so as to minimize the fuel cost of thermal units. The problem requires that a given amount of water be used in such a way so as to achieve this objective, which is usually much more complex than the scheduling of all thermal system. This is because if water available is used up in the present interval there will not be any water for the next interval increasing this way the future operation costs. Both electrically and hydraulically coupled hydro plants themselves are difficult to co-ordinate with the thermal generation system to obtain minimum total system cost. Diverse hydraulic and thermal constraints should be satisfied in the hydrothermal scheduling problem. These constraints consist of load balance, operating capacity limits of the hydro and thermal units, water discharge rate, upper and lower bounds on reservoir volume, water spillage and hydraulic continuity restrictions. Additional constraints such as the need to satisfy activities including: flood control, irrigation, navigation, fishing, water supply, recreation, etc., may also be considered.

The problem is solved by using some conventional methods like Gradient Search [1]. In conventional methods simplifying assumptions are made to make the solution tractable. This is impractical. So, non conventional methods are adopted for solving this nonlinear problem. Methods like Genetic Algorithm [2,3,4], Evolutionary programming[5,6,7] simulated annealing[8], neural network based techniques [9] are implemented. Simulated annealing requires more computational time and the tuning of its parameters is not an easy task. Most of the GA parameters are set after considerable experimentation, and it is the lack of a solid theoretical basis for their setting which is one of the main drawbacks of the GA method. Theoretical research is continuing on the appropriate choice of GA parameters. Also the encoding and decoding schemes demand a higher computational time and computer memory. Among the existing methods EP seems to have given the best results. But the computational time is not negligible. To overcome these problems a novel approach of scheduling the hydro thermal generation using the particle swarm optimization algorithm is proposed [10,11,12]. In this method the number of parameters to be tuned is less compared with other non conventional methods. PSO does not have genetic operators like crossover and mutation. Objective function is directly used as fitness function to guide the search in PSO, making it easy to handle non-linear and non-differentiable optimization problem.
The complexity analysis of problem-dimension using PSO has been reported for three well-known benchmark functions, DeJong, Rosenbrock and Rastrigin in [13]. Hybrid Differential Evolution [14] approach is a simple population based stochastic function method and has been extended from the original algorithm of differential evolution [15]. A coevolutionary hybrid differential evolution for mixed-integer optimization problems which combines a local heuristic (acceleration) and a widespread heuristic (migration) to promote the search for a global optimum was proposed[16]. The application of a robust searching hybrid differential evolution method for optimal reactive power planning in large scale distribution systems was presented.[17].A modified differential evolution (MDE) algorithm, for solving short-term hydrothermal scheduling problem is presented[18].These evolutionary computation algorithms has applications not only in the area of Power Systems but also in the areas such as cryptanalysing block ciphers[19].

In this paper a hybrid differential evolution algorithm and a particle swarm optimization algorithm are developed for the short term hydrothermal scheduling problem. To demonstrate its applicability a test system is considered. The solutions obtained are compared with EP, GA, SA and gradient search methods.

2 Problem Formulation

The objective function and associated constraints of the hydrothermal scheduling problem are formulated as follows

2.1 Objective Function

The objective function which is the total fuel cost for running the thermal system to meet the load demands in a schedule horizon is given by

$$f_{total} = \sum_{j=(1,n)} n_j f_j (PT_j) \qquad (1)$$

2.2 Constraints

The problem is subject to a variety of constraints both static and dynamic.

Equality Constraints:

The equality constraints are the power balance constraints, total water discharge constraint and the reservoir volume constraints. The power balance constraints are described as

$$PD_j = PT_j + PH_j - PL_j \quad for\ j = 1, 2,..,z \qquad (2)$$

where, the electric loss between the hydro plant and the load PL_j is given by,

$$PL_j = k (PH_j)^2 \qquad (3)$$

According to the above constraint the total generation in any interval has to equal demand in that particular interval taking into consideration the generation losses which are here a function of hydro generation alone. The constant head operation is considered and the water discharge rate, q_j, is assumed to be a function of hydro plant generation, PH_j, as described below.

$$q_j = g\ (PH_j) \quad (4)$$

The total water discharge constraint is given by

$$q_{total} = \sum_{j=(1,z)} n_j q_j \quad (5)$$

In the case of a storage reservoir with a given initial volume and a given final volume, the reservoir volume constraints are expressed as,

$$V_j = V_{j-1} + n_j\ (r_j - g(PH_j) - s_j) \quad for\ j = 1,2,3\ldots\ldots\ldots,z \quad (6)$$

Inequality Constraints:

The inequality constraints are the operation limits of the equivalent thermal generator and those of the hydro plant and the reservoir storage limits. These constraints are expressed for j =1, 2,…, Z as below.

$$PT_{min} \le PT_j \le PT_{max}$$
$$PH_{min} \le PH_j \le PH_{max} \quad (7)$$
$$V_{jmin} \le V_j \le V_{jmax}$$

In order to obtain the optimum hydrothermal generation schedule in all the intervals of the schedule horizon, the objective function in Eq.1 is minimized subject to all equality and inequality constraints described above.

3.The Hybrid Differential Evolution Technique

Hybrid Differential Evolution [14] approach is a simple population based stochastic function method and has been extended from the original algorithm of differential evolution [15]. This method is used to solv-e unconstrained nonlinear, non-smooth and non-differentiable optimization problems. The basic operations of HDE are as given below:

Step 1:Representation and Initialization

HDE is a parallel direct search algorithm that utilizes N_P vectors of decision variables **x** in the non-linear programming problem, i.e $X^G=\{x_i^G,\ i=1,\ldots\ldots, N_P\}$,as a population in generation G. For convenience, the decision vector (chromosome), $\mathbf{x}_i$, is represented as $(x_{1i}\ ,\ldots\ x_{ji}\ ,\ldots\ x_{Ni}\ )$.Here, the decision variable (gene), x_{ji} , is direct coded as a real value within its corresponding lower-upper bounds. The initialization process generates N_P individuals $\mathbf{x}_i$ randomly, and has to cover the entire search space uniformly in the form

$$\mathbf{x}_i^0=\mathbf{x}^L+\rho_i\,(\mathbf{x}^U - \mathbf{x}^L)\ ,\ i=1,\ 2,\ldots\ldots, N_P \quad (8)$$

where ρ_i is a vector of a random number in the range [0,1].The N genes of each individual are the powers Generated by each generator satisfying the inequality (Generation limits) and equality (power balance) Constraints and hence form a feasible solution.

Step 2: Mutation

Pairs of individual vectors from step 1 are chosen at random. A mutant individual is generated by,

$$\mathbf{u}_i^{G+1}=\mathbf{x}_p^G+\rho_m\,(\mathbf{x}_j^G-\mathbf{x}_k^G),\ i=1,2\ldots\ldots,N_P \quad (9)$$

where random indices $p,j,k \in \{1,2,\ldots.. N_P\}$ are integer values and mutually different. The mutation factor ρ_m is a real-valued random number between zero and one.

Step 3: Crossover Operation

The crossover operation is performed to increase the local diversity of the population. This operation reproduces an offspring at the next generation. The newly mutant individual in $\boldsymbol{u}_i^{G+1}$ in Eq.(9) and the current individual $\boldsymbol{x}_j^G$ are chosen by a binomial distribution to perform the crossover operation. In this operation, each gene of the ith individual is reproduced from the mutant vectors $\boldsymbol{u}_i^{G+1} =(u_{1i}^{G+1}, u_{2i}^{G+1},\ldots\ldots u_{Ni}^{G+1})$ and the current individual $\mathbf{x}_i^G =(x_{1i}^G, x_{2i}^G, \ldots x_{Ni}^G)$ as follows :

$$u_{ji}^{G+1}= \begin{cases} x_{ji}^G & \text{, if a random number} > C_R \\ u_{ji}^{G+1} & \text{, otherwise;} j=1,\ldots..N,\ \ i=1,\ldots..,N_P \end{cases} \quad (10)$$

Where the crossover factor $C_R \in [0,1]$ is a constant and has to be set by the user.

Step 4: Selection and Evaluation

The offspring is compared with its parent and it replaces the parent if its fitness is higher. Else the parent is retained. Here the fitness function is the objective function of the various equations is section 2. Two selection steps are performed in this evaluation expression. The first step is a one-to-one competition, and the next step is to select the best individual in the population. These two steps are expressed in the forms

$$\hat{X}_i^{G+1} =argmin\{ f(\mathbf{x}_i^G), f(u_i^{G+1}) \},\ i=1,\ldots\ldots.,N_p \quad (11)$$

$$\hat{X}_b^{G+1} = argmin\{f(\mathbf{x}_i^{G+1}),\ i=1,\ldots\ldots\ldots., N_p\} \quad (12)$$

where argmin means the argument of the minimum. From Eqs. (11) and (12), the best individual $\mathbf{x}_b^{G+1}$ can be kept at each generation.

Step 5: Accelerated Operation

An accelerated operation and a migration operation are used as a trade-off. The accelerated operation is used to speed up the convergence, whereas the migration operation is used to evade the local minima. If the best individual is no longer improved by mutation and crossover, the gradient of the objective function (∇f) obtained by finite difference is applied to push the best individual to a better point by the steepest descent method. The acceleration operation is therefore expressed as

$$\mathbf{x}_b^{N} = \mathbf{x}_b^{G+1} - \rho_a \nabla_X f(\mathbf{x}) / \mathbf{x}_b^{G+1} \quad (13)$$

where $\mathbf{x}_b^{N}$ is the newly best solution .The continuous gradient of the objective function, $\nabla_X f(\mathbf{x})$ **,** can be approximately calculated with a finite difference method. The step size $\rho_a \in [0,1]$ is judiciously chosen for proper convergence. The objective function value $f(\mathbf{x}_b^{N})$ is then compared with $f(\mathbf{x}_b^{G+1})$.If the descent property is obeyed, i.e.,

$$f(\mathbf{x}_b^{N}) < f(\mathbf{x}_b^{G+1}) \quad (14)$$

the new individual $\mathbf{x}_b^{N}$ is added into this population to replace the worst individual. On the other hand, if the descent property fails, the step size ρ_a is adjusted. The descent method is repeated to obtain $\mathbf{x}_b^{N}$ until $\rho_a \nabla_X f$ becomes sufficiently small or an iteration limit is exceeded. Consequently, the best fitness $f(\mathbf{x}_b^{N})$ should be at least equal to or smaller than $f(\mathbf{x}_b^{G+1})$.

Step 6: Migration Operation

In order to greatly increase the exploration of the search space and decrease the selection pressure of a small population, a widespread search heuristic called migration is introduced to generate a newly diversified population of individuals. The newly migrant individuals are generated on the basis of the best individual $\mathbf{x}_b^{G+1} = (x_{1b}^{G+1}, \ldots\ldots x_{Nb}^{G+1})$ by using non-uniformly random choice. New genes of the ith individual are therefore generated by

$$x_{ji}^{G+1} = \begin{cases} x_{jb}^{G+1} + \rho(x_j^{L} - x_{jb}^{G+1}), & \text{if a random number} < (x_{jb}^{G+1} - x_j^{L}) / (x_j^{U} - x_j^{L}) \\ x_{jb}^{G+1} + \rho(x_j^{U} - x_{jb}^{G+1}), & \text{otherwise} \end{cases} \quad (14)$$

where,

$j=1, \ldots N$

$i=1, \ldots\ldots\ldots N_p-1$

Where ρ is a random number in the range [0,1].
The migration operation of HDE is performed only if a measure of population diversity does not match the desired tolerance. Hence we use a measure t_t ρ_m defined as follows.

$$\rho_m = \Big(\sum_{\substack{i=1 \\ i \neq b}}^{N_P} \sum_{j=1}^{N} \eta_{ji}\Big) / (N(N_P - 1)) < \varepsilon_1 \quad (16)$$

Where

$$\eta_{ji} = \begin{cases} 1, & \text{if } \left| (u_{ji} - u_{jb}) / u_{jb} \right| > \varepsilon_2 \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

where ε_1 *and* ε_2 are the desired tolerance for the group diversity and gene diversity with respect to the best individual. Here η_{ji} is defined as an index of gene diversity. Its value is zero if the jth gene of the ith individual closely clusters with the jth gene of the best individual.
The migration operation is performed only if the degree of population diversity is smaller than the desired tolerance ε_1. From (16) it is inferred that the degree of population diversity is between zero and one. A value of zero implies that all genes cluster around the best individual. Conversely, a value of 1 indicates that the current candidate individuals are a completely diversified population. The desired tolerance for population diversity is accordingly assigned within this region. Zero tolerance implies that the migration is switched off whereas a tolerance of 1 implies that the migration operation is performed at every generation.
A flow chart for a hybrid differential evolution algorithm is shown in Fig.1.

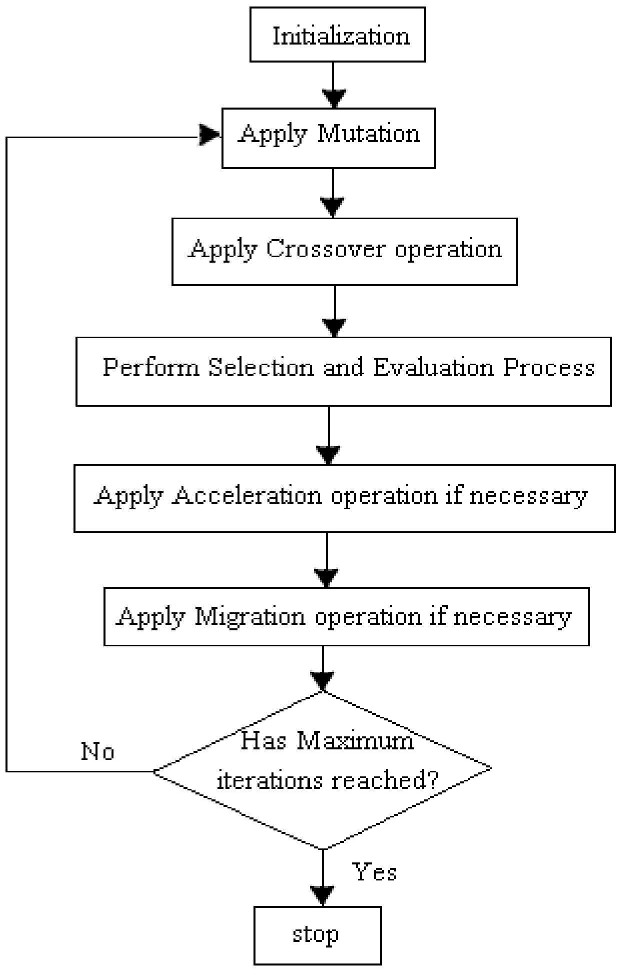

Fig.1 Hybrid differential evolution algorithm

4. Overview of Particle Swarm Optimization

Kennedy and Eberhart first introduced PSO in the year 1995 [20]. PSO is motivated from the simulation of the behaviour of social systems such as fish schooling and birds flocking [21]. The PSO algorithm requires less computation time and less memory because of the simplicity inherent in these systems. The basic assumption behind the PSO algorithm is, birds find food by flocking and not individually. This leads to the assumption that information is owned jointly in flocking. PSO is distinctly different from other evolutionary optimization methods in that it does not use the filtering operation (such as crossover and/or mutation) and the members of the entire population are maintained through the search procedure.

PSO algorithm for *N*-dimensional problem formulation can be described as follows. Let *P* be the 'particle' co-ordinates (position) and *V* its speed (velocity) in a search space. Consider *i* as a particle in the total population (swarm). Now the i^{th} particle position can be represented as $P_i = (P_{i1}, P_{i2}, P_{i3}, \ldots P_{iN})$ in the *N*-dimensional space. This is stored as the best previous position of the i^{th} particle and is represented as $Pbest_i = (Pbest_{i1}, Pbest_{i2}, \ldots Pbest_{ij})$. All the *Pbest*s *are* evaluated by using the objective function of the problem. The best particle among all *Pbest* s is represented as *gbest*. The velocity of the i^{th} particle is represented as $V_i = (V_{i1}, V_{i2} \ldots V_{ij})$. The modified velocity of each particle can be calculated using the information of

(i) The current velocity
(ii) The distance between the current position and *Pbest* and
(iii) The distance between the current position and *gbest.*

This can be formulated as an equation as

$$V_{ij}^{(k+1)} = w * V_{ij}^{(k)} + c_1 * rand_1 * (Pbest_{ij} - P_{ij}^{(k)}) + c_2 * rand_2 * (gbest_i - P_{ij}^{(k)}) \quad (18)$$

$$P_{ij}^{(k+1)} = P_{ij}^{(k)} + V_{ij}^{(k+1)}, \; i = 1, 2, \ldots I, \text{ and } j = 1, 2, \ldots N \quad (19)$$

Where

N	number of dimensions in a particle
I	number of particles
w	inertia weight factor
c_1, c_2	acceleration constant
$rand_1$, $rand_2$	uniform random value in the range [0, 1]
$V_{ij}^{(k)}$	velocity of j^{th} dimension in i^{th} particle, $V_j^{min} \le V_{ij}^{(k)} \le V_j^{max}$
$P_{ij}^{(k)}$	current position of the j^{th} dimension in i^{th} particle at iteration k

The use of linearly decreasing inertia weight factor w has provided improved performance in all the applications. Its value is decreased linearly from about 0.9 to 0.4 during a run. Suitable selection of the inertia weight provides a balance between global and local exploration and exploitation, and results in less iteration on an average to find a sufficiently optimal solution. Its value is set according to the following equation:

$$w = w_{max} - \frac{w_{max} - w_{min}}{k_{max}} * k \quad (20)$$

Where w_{max} and w_{min} are both random numbers called initial weight and final weight respectively

k_{max} the maximum iteration number
k the current iteration number

In equation (18) the first term indicates the current velocity of the particle, second term represents the cognitive part of PSO where the particle changes its velocity based on its own thinking and memory. The third term represents the social part of PSO where the particle changes its velocity based on the social-psychological adaptation of knowledge [20].

5. Implementation of PSO method

PSO algorithm presents a quick solution to the short term hydro thermal scheduling problems. Its implementation consists of the following steps

A flow chart for a particle swarm optimization is shown in Fig.2.

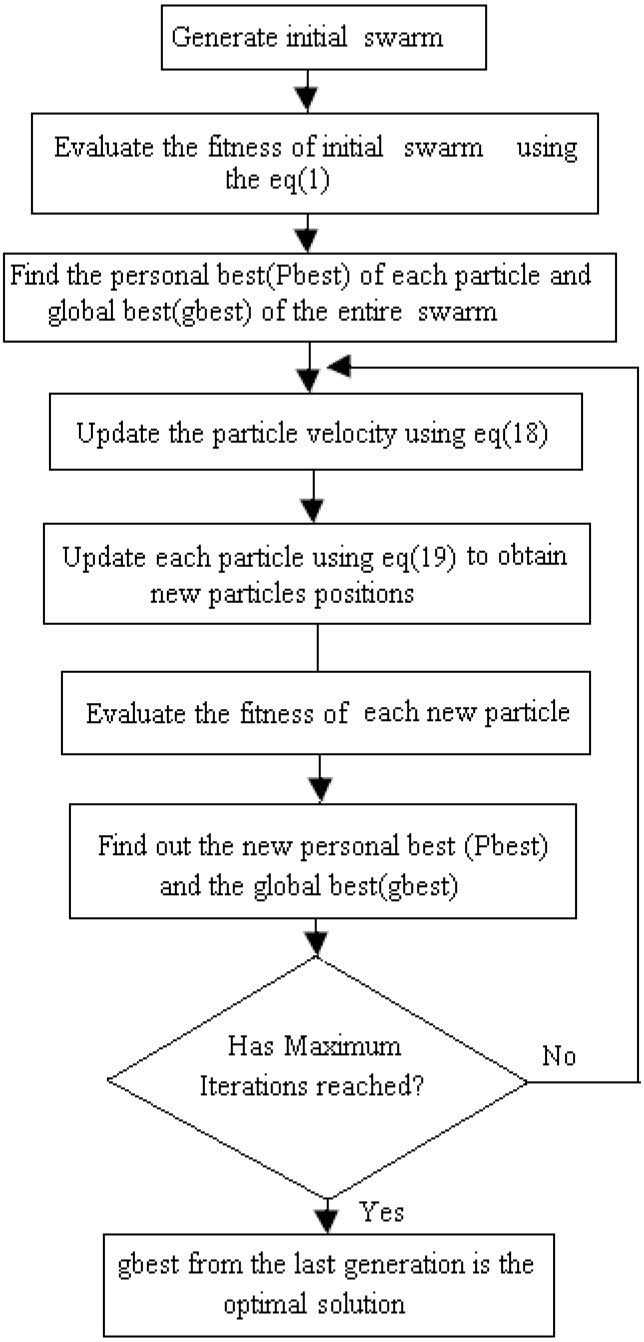

Fig.2 Particle swarm optimization algorithm.

PSO implementation consists of the following steps
Step 1:
The particles are randomly generated between the maximum and the minimum operating limits of the generators. For example, if there are N units, the i[th] particle is represented as follows:

$$P_i = (P_{i1}, P_{i2}, P_{i3}, \ldots P_{iN})$$

Step 2:
The particle velocities are generated randomly in the range [$-V_j^{max}$, V_j^{max}].

The maximum velocity limit in the j[th] dimension is computed as follows:

$$V_j^{\max} = \frac{P_{j\max} - P_{j\min}}{R} \qquad (21)$$

Where R is the chosen number of intervals in the j[th] dimension.For all the examples tested using the PSO approach, V_j^{max} was set at 10 – 20 % of the dynamic range of the variable on each dimension.

Step 3
Objective function values of the particles are evaluated using Eq. (1). These values are set as the Pbest value of the particles.

Step 4
The best value among all the Pbest values is identified as gbest.

Step 5
New velocities for all the dimensions in each particle are calculated using equation (18).

Step 6
The position of each particle is updated using equation (19).

Step 7
The objective function values are calculated for the updated positions of the particles. If the new value is better than the previous Pbest, the new value is set to Pbest.

Step 8
The current gbest among the updated Pbest particles is determined.

Step 9
The current gbest is compared with the previous gbest and is updated if it is less than the previous gbest. If the stopping criterion of maximum number of iterations is reached, the particle represented by gbest is the optimal solution. Otherwise the procedure is repeated from step (5).

6. Hydro Thermal Scheduling Evaluation Sequence

1. The hydro plant water discharge rates are taken as decision variables instead of hydro plant power generation, since handling of the water balance constraint is difficult.
2. Volume at the end of each interval is calculated using eq. (6). With the known

discharge rate hydro generation is calculated using eq. (4) and thermal generation values are found by solving eq. (2).

3. The cost of thermal power generation is calculated using eq. (1). This is the fitness value for each individual in the population.
4. Modify the position of each individual by using eq. (19) in case of PSO or by using the operations of HDE namely mutation, crossover, acceleration and migration in case of HDE.
5. According to different constraint eqs. (5), (6) and (7) the constraint violations VIO_k are determined. The fitness of the individuals is computed using eq. (22).

$$C_{total}=f_{total}+\sum_{k=1}^{NC}\lambda_k\ (VIO_k)^2 \qquad (22)$$

   Where C_{total} is the evaluation value of individuals violating the constraints. NC is the number of constraints, VIO_k is the amount of violation of constraint k and λ_k is the penalty multiplier for the violated constraint k.
6. Steps 2 to 5 are repeated untill the maximum number of generations are reached. The minimum solution obtained during the evaluations is considered as the optimal solution to the problem.

7. Simulation Results:

To illustrate the PSO based hydrothermal scheduling algorithm, we consider a test system [1] comprising of a hydro plant and an equivalent thermal plant. The load duration is for six intervals and each interval is of 12 hours duration. The demand for the six intervals is given in Table 1.

Table 1. Demand for successive intervals

INTERVAL	1	2	3	4	5	6
LOAD(MW)	1200	1500	1100	1800	950	1300

The fuel cost function for the thermal plant is
FC= 575+ (9.2 x P_s) + (0.00184 x P_s^2) for 150 > P_s >1500.

The hydro plant power generation relationship is given as
q = 330 + (4.97 x P_H) for $0 \le P_H \le 1000$
q = 5300 + (12 x (P_H - 1000)) + (0.05 x (P_H - 1000)^2) for $1000 < P_H < 1100$
The hydro plant data is given in Table 2.The operating limits of the hydro plant: [0, 1100].The operating limits of the thermal plant: [150, 1500]

Table 2. Hydro Plant Data

V_{min}	V_{max}	q_{min}	q_{max}	V_0	V_6	R
60 000	120 000	330	7000	100 000	60 000	2000

The program is developed using Matlab 6.5 and run on a PC Pentium 4(2.00 GHz, 256MB RAM).The optimal control parameters used in HDE are listed in Table 3 .The PSO parameters selected for the solution obtained are given in Table 4.

Table 3. Best hybrid differential evolution parameters

HDE Parameters	Value
No of individuals , N_p	30
Max generation number	300
Mutation factor, ρ_m	0.5
Crossover factor, C_R	0.5
Time of convergence, sec	1.172

Table 4. Best particle swarm optimization parameters

PSO Parameteres	Value
No of particles, I	30
Acceleration factors, c_1,c_2	1.5
Max iterations, k_{max}	300
Inertia factor, w_{min}	0.1
Inertia factor, w_{max}	1.0
Time of convergence, *sec*	0.751

Table 5. Simulation Results of HDE and PSO and their comparision with other Optimization techniques

Method	Interval	Ps	P_H	V	q	Cost
Gradient Search	1	903.11	296.89	102 333.3	1805.56	709 877.38
	2	889.22	610.78	85 946.7	3365.56	
	3	893.65	206.35	93 680	1355.56	
	4	899.26	900.74	60 000	4806.67	
	5	806.25	143.75	71 466.7	1044.44	
	6	771.72	528.28	60 000.1	2955.56	
Simulated Annealing	1	893.73	306.27	101 773.81	1852.18	709 874.36
	2	895.24	604.76	85 746.05	3335.65	
	3	884.21	215.68	92 922.88	1404.93	
	4	912.87	887.78	60 015.62	4742.27	
	5	781.87	168.13	70 028.41	1165.6	
	6	795.83	504.17	60 000	2835.7	
Genetic Algorithm	1	896.86	301.14	10 196.94	1836.59	709 863.56
	2	897.15	602.85	86 046.83	3326.18	
	3	893.85	206.15	93 791.98	1354.57	
	4	897.38	902.62	60 000	4816.01	
	5	794.45	155.55	70 763.09	1103.08	
	6	783.52	516.48	60 000.01	2896.92	
Evolutionary Programming	1	895.57	304.43	101 883.96	1843.00	709 862.06
	2	897.69	602.31	86 002.19	3323.48	
	3	895.19	204.81	93 827.48	1347.89	
	4	896.79	903.21	60 000.15	4818.94	
	5	788.70	161.30	70 420.01	1131.68	
	6	789.27	510.73	60 000	2868.33	
Particle Swarm Optimization	1	893.99	306.00	101 790.15	1850.82	709 862.07
	2	898.11	601.88	85 933.59	3321.38	
	3	895.70	204.29	93 789.29	1345.35	
	4	897.42	902.57	60 000.00	4815.77	
	5	787.74	162.25	70 363.04	1136.41	
	6	790.22	509.77	60 000.00	2863.58	
Hybrid differential evolution	1	896.61	303.38	101946.12	1837.82	709862.09
	2	895.11	604.88	84343.24	3336.27	
	3	896.44	203.55	92243.13	1341.67	
	4	897.06	902.93	60 000.00	4817.56	
	5	788.94	161.05	70 434.79	1130	
	6	789.02	510.97	60 000.00	2869.56	

The parameters are tuned such that convergence is achieved at much faster rate compared with other methods. A population of 30 particles is considered in both the algorithms. Each program is run for a maximum of 300 iterations. So, a total of 30*300= 9000 trails are run for the optimum solution. The program is run 100 times for each algorithm. The efficiency of the algorithms can be judged from the results obtained.

Table 5 shows the comparison of the results obtained by the proposed Hybrid Differential Evolution and Particle Swarm Optimization techniques with those obtained by the conventional gradient search method and the non-conventional methods such as simulated annealing, genetic algorithm, evolutionary programming. From the Table it is observed that HDE and PSO give the same solution as obtained by other. This solution is the global optimum**.**

Fig.3 shows the convergence characteristics of both HDE and PSO methods for the test system considered. From the figure it is observed that the convergence is smooth and faster in case of PSO.

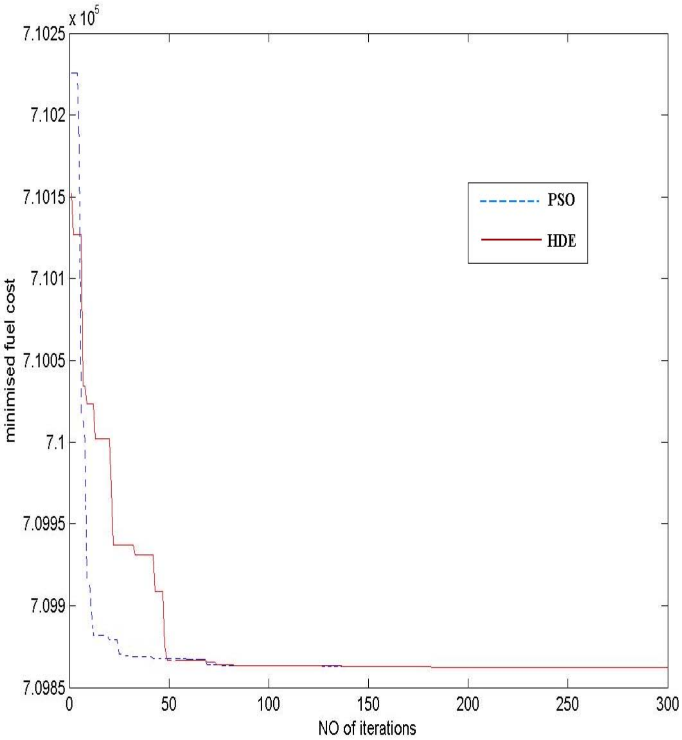

Fig.3. Convergence Characteristics of HDE and PSO algorithms for short term hydro thermal scheduling

8. Conclusion

The applicability of HDE and PSO algorithms for solving short term hydrothermal scheduling problems is demonstrated. The results obtained for the example problem considered in this paper indicate that highly near optimum solutions can be achieved when compared to simulated annealing, genetic algorithm, evolutionary programming and gradient search methods. For the same number of trials the time needed for PSO is lesser than that required in other methods. This proves the potential of HDE and PSO to find a more nearly optimal solution to the hydrothermal scheduling problems. In addition, the superior features of the algorithms are i) Simple and efficient. ii) Any number and types of constraints can be easily accommodated. iii) suitable for solving any type of objective function (irrespective of the shape). iv) Reduced computing time. v) Smooth and fast convergence and vi) Better quality solutions. The evolutionary algorithms are still in the development stage and its implementation for online applications needs further research. In order to fully exploit the potential of PSO for solving large-scale hydrothermal scheduling problems with cascaded reservoirs, the number of iterations required and hence the time for convergence has to be reduced, and this remains to be tested.

Acknowledgement

The authors thank the management of Vellore Institute of Technology University, Vellore, India for providing the necessary facilities and encouragement to carry out this work.

References

[1] A.J.Wood, B.F.Wollenberg, *Power Generation, Operation and Control*, John Wiley & Sons, New York, 1984.

[2] S.O.Orero and M.R.Irving, A genetic algorithm modelling framework and solution technique for short term optimal hydrothermal scheduling, *IEEE Transactions on Power Systems,*Vol.13, No. 2,May 1998, pp. 501-518.

[3] Christoforos E. Zoumas, Anastasios G. Bakirtzis, John B. Theocharis, and Vasilios Petridis, A Genetic Algorithm Solution Approach to the Hydrothermal Coordination Problem, "*IEEE Transactions on Power Systems*, vol. 19, No. 2, may 2004, pp.1356-1364.

[4] Esteban Gil, Julian Bustos, and Hugh Rudnick, "Short-Term Hydrothermal Generation Scheduling Model Using a Genetic Algorithm", *IEEE Transactions on Power Systems*, vol. 18,

[5] P.-C. Yang, H.-T. Yang and C.-L. Huang Scheduling short-term hydrothermal generation using evolutionary programming techniques, *IEEE, Proc.Inst. Elect.Eng.*, vol.143, No. 4, July 1996, pp. 371–376.

[6] P. K. Hota, R. Chakrabarti, and P. K. Chattopadhyay, "Short-term hydrothermal scheduling through evolutionary programming technique", IEEE, EPSR, vol.52, No. 2, November 1, 1999, pp.189-196.

[7] Nidul Sinha, R. Chakrabarti, and P. K. Chattopadhyay, Fast Evolutionary Programming Techniques for Short-Term Hydrothermal Scheduling, *IEEE Transactions on Power Systems*, vol. 18, no.1, February 2003, pp. 214-220

[8] K.P.Wong, Y.W.Wong, Short-term hydrothermal scheduling. Part 1: simulated annealing approach, *IEE Proc. C* 141 (1994), pp.497–501.

[9] R.Naresh and J.Sharma, Two-Phase Neural Network Based Solution Technique For Short Term Hydrothermal Scheduling*, IEE Proc-Gener. Transm. Distrib..*, Vol. 146, No. 6, November 1999,pp. 651-657.

[10] Jong-Bae Park, Ki-Song Lee, Joong-Rin Shin, and Kwang Y. Lee, A Particle Swarm Optimization for Economic Dispatch with Non smooth Cost Functions, *IEEE Transactions On Power Systems,* Vol. 20, No.1, February 2005, pp. 34-43.

[11] Zwe-Lee-Gaing, "Particle Swarm Optimization to solve the Economic Dispatch considering generation constraints," IEEE, EPSR, vol. 18, No.3, August 2003, pp.1187–1195,

[12] Binghui Yu, Xiaohui Yuan and Jinwen Wang, Short-term hydro-thermal scheduling using particle swarm optimization method, *Energy conversion and Management 48(2007) 1902-1908.*

[13] Buthainah S. AL-Kazemi and Sami J.Habib, Complexity Analysis of Problem-Dimension Using PSO, Proceedings of the 7th WSEAS International Conference on Evolutionary Computing, Cavtat, Croatia, June 12-14, 2006 (pp45-52).

[14] Feng-Sheng Wang and Horng-Jhy Jang, Parameter Estimation of a Bioreaction Modelby Hybrid Differential Evolution, *Evolutionary Computation(2000), Proceedings of the 2000 Congress*, pp.410417.

[15] Rainer Storn and Kenneth Price,Minimizing the real functions of the ICEC'96 contest by Differential Evolution, *IEEE Inernational Conference on Evolutionary Computation (1996)*, pp.842-844.

[16] Yung-Chien Lin, Kao-Shing Hwang and Feng-Sheng Wang, Co-Evolutionary Hybrid Differential Evolution for Mixed Integer Optimization Problems, *Engineering Optimization (2001)*, vol.00, pp.1-20.

[17] Chung-Fu Chang, Ji-Jen Wong, Ji-Pyng Chiou Ching-Tzong Su, Robust searching hybrid differential evolution method for optimal reactive power planning in large-scale distribution systems, *Electrical Power Systems Research*, April 2006.

[18] L.Lakshminarasimman and S.Subramanian, Short-term scheduling of hydrothermal power system with cascaded reservoirs by using modified differential evolution*, IEE proc-Gener. Transm. Distrib.* vol.153. No.6. November 2006.

[19] Nalini N and Ragavedra rao G,Experiments on crptanalysing block ciphers via evolutionary computation paradigms, Proceedings of the 7th WSEAS International Conference on Evolutionary Computing, Cavtat, Crotia, June 12-24,2006(pp20-27).

[20] Kennedy J. and Eberhard R.C., Particle Swarm Optimization, *Proc. of IEEE Int'l Conf. on Neural Networks, Piscataway, NJ, USA*, 1995, pp.1942-1948.

[21] Eberhart.R.C, Shi.Y, Particle Swarm Optimization: Developments, *Applications and Resources. Proceedings of the congress on Evolutionary Computation.2001*, Vol.1:81-86.