PARTICLE SWARM OPTIMIZATION AND ITS VARIANTS FOR SHORT TERM HYDROTHERMAL SCHEDULING

Salman Fakhar
University of Lahore Thokar Lahore

Muhammad Asghar Saqib
University of Engineering and Technology, Lahore

PARTICLE SWARM OPTIMIZATION AND ITS VARIANTS FOR SHORT TERM HYDROTHERMAL SCHEDULING

Muhammad Salman Fakhar[1],Syed Abdul Rahman Kashif[2],Muhammad Asghar Saqib[3]
[1,2,3]University of Engineering and Technology, Lahore 54890, Pakistan
Email:salman.fakhar@ee.uol.edu.pk, abdulrahman@uet.edu.pk, saqib@uet.edu.pk

ABSTRACT-*Short term hydrothermal scheduling has been a hot topic of discussion among the power system operation and control engineers for its great economic importance. A lot of work has been reported till date to make the dispatch of the thermal and hydal generating units in such a way that the cost of operation is minimized. Many conventional non-heuristic and non- conventional heuristic optimization techniques have been applied to the multi modal and non-linear optimization problem of dispatching the hydrothermal units economically. Literature reveals that among all the optimization techniques applied so far, Particles Swarm Optimization (PSO) and its variants have performed the best. This paper presents a review on the implementation of the famous heuristic particle swarm optimization technique and its different variants, i.e. Canonical PSO (CPSO), Improved PSO (IPSO), Self-Organizing Hierarchical PSO(SOHPSO), Global Vision of PSO with Inertia Weight (GWPSO), Local Vision of PSO with Inertia Weight (LWPSO), Global Vision of PSO with Constriction Factor (GCPSO) and Local Vision of PSO with Constriction Factor (LCPSO)on the hydrothermal scheduling problem. Moreover, the paper discusses a new variant of CPSO known as fully informed particle swarm optimization (FIPSO) that is applied on the short term hydrothermal scheduling problem*.

Index Terms: Particle swarm optimization, short term hydrothermal scheduling, and economic dispatch.

1. INTRODUCTION

Hydro-thermal scheduling is an economic dispatch problem in which simultaneous dispatching of hydal and thermal units is so applied that the cost of operation is minimized. This cost is mainly the fuel cost of the thermal generating units [1,2]. This problem is accomplished using either classical optimization or heuristic optimization algorithms [3,4]. The conventional methods such as Langrage multiplier, dynamic programing and gradient search have already been deployed to solve the economic dispatch problem as discussed in reference [4]. According to references [3,4], the short term hydrothermal scheduling has been done using stochastic optimization techniques like evolutionary programming and its variants such as fast evolutionary programming and hybrid technique, simulated annealing and parallel simulated annealing, a diploid genetic approach and genetic algorithm, honey bee mating optimization, and different variants of Particle Swarm Optimization.

The particle swarm optimization (PSO) method has recently been applied in its canonical form in many power system operation and control applications. There exits different variants of PSO, which provides better optimization as compared to canonical form has been discussed in literature for their effectiveness. The conventional techniques have failed to search the global optimum solution of multi-model (multiple optima) non-linear optimization problems, therefore heuristic algorithms have been used in power system economic dispatch problems. Particle Swarm Optimization and its Variants have proved best among the different genetic algorithms in such problems for it is quite realistic, its convergence speed is quite high, and it has the best capabilities to get rid of the local optima and direct towards the global optima. [5]

References [3-10] give the literature on the various implementations of PSO and its variants on the hydrothermal scheduling problem. This review paper will discuss the methodologies utilized in [3-10] quite handsomely so that a new researcher may find all the works in this dimension in one paper. References [10] and [11] introduce a new variant of PSO known as Fully Informed Particle Swarm Optimization (FIPSO). This paper also gives the results for the optimization performed using (FIPSO) that is implemented by the author of this paper and it is interesting to know that this implementation has outperformed all the previous implementations.

2. HYDROTHERMAL SCHEDULING PROBLEM

According to the references [1] and [2], in power systems' operation and control, the short term hydrothermal scheduling problem is described as the dispatch of the thermal generating units with such a power output that the simultaneously and adding hydro units produce that much power that equalizes the load demand and thus the operating cost of the thermal unit, which is mostly due to the fuel cost, is minimized. Short term means that the scheduling period is not more than a week.

Mathematically, the hydrothermal scheduling problem can be defined as;

$$\min(f) = \sum_{j}^{N} n_j F_j \quad (1)$$

Subject to the water discharge constraints;

$$\sum_{j}^{N} n_j D_j = D_{tot} \quad (2)$$

The Power balance constraints are;

$$P_{load} + P_{loss} = P_{hydal} + P_{thermal} \quad (3)$$

Where, n is the number of hour in the jth scheduling interval with comprises on twelve hours in our case. F_j is the operating cost of the interval j. In hydro-generation the discharge rate of water is a major concern as usually the major purpose is the irrigation; therefore the water discharge constraint considered is given by the following equation:

P_{loss} is the transmission loss and is given by;

$$P_{loss} = f(P_{hydal}) \quad (4)$$

Hydro generation is the function of discharge rate only;

$$P_{hydal} = f(\mathrm{D_j}) \quad (5)$$

Where

$$\begin{cases} D_{min} < D_j < D_{max} & (Water\ discharge\ limits) \\ P_{thermal,min} < P_{thermal,j} < P_{thermal,max} & (Thermal\ generation\ limits) \\ P_{hydal,min} < P_{hydal,j} < P_{hydal,max} & (Hydro\ generation\ limits) \end{cases} \quad (6)$$

The volume of the reservoir has the following constraints.

$$V_j = V_{j-1} + n_j(R_j - D_j - S_j) \quad (7)$$

Where,

'R_j' is the water inflow rate at j[th] interval

'S_j' is water spillage rate in the j[th] interval

'V_j' is the volume of the reservoir at the j[th] interval and

'D_j' is the discharge rate at the j[th] interval.

The volume of the reservoir must follow the following inequality constraints.

$$V_{min} < V_j < V_{max} \quad (8)$$

It can be understood from the above statements that the main objective is to minimize the production cost of hydrothermal energy while meeting the hydro and thermal units' constraints.

3. PSO AND ITS VARIANTS' MODELS FOR SHORT TERM HYDROTHERMAL SCHEDULING

References [3] – [9] give the different models of Particle Swarm Optimization that have been implemented on the above mentioned short term hydrothermal scheduling problem. A brief review of these references is given in an order below.

3.1. Short Term Hydrothermal Scheduling Using Particle Swarm optimization(CPSO)

Reference [3] and [4] use the canonical version of particle swarm optimization, i.e. (CPSO) to solve the short term hydrothermal scheduling problem. In the very problem, there are four candidates of being the potential particles, i.e. the hydal power, the thermal power, the discharge rate and the volume of water in the reservoir. Reference [3] utilizes turn by turn all the candidates being reference, however, it narrates that the volume of water in the reservoir is the best candidate for being the particle. The particle variable becomes then the independent variable while the remaining three candidates become the dependent variables in the problem. The equation utilized is of CPSO and is given as;

$$\begin{cases} V_j^{k+1} = W.V_j^k + c_1 rand(0,1)(P_{best} - x_j^k) + c_2 rand(0,1)(G_{best} - x_j^k) \\ x_j^{k+1} = x_j^k + K.V_j^{k+1} \end{cases} \quad (9)$$

Here, V_j^k is the velocity of j[th] particle at the k[th] iteration, x_j^k is the position of the j[th] particle at the k[th] iteration. *W* is the weight parameter, c_1 and c_2 are the acceleration coefficients and are equal to 2.5. *rand(0,1)* is the uniform random number generator between 0 and 1. P_{best} and G_{best} are the local and global best positions of the particles found till the k[th] iteration.The main steps of this algorithm are

1. Start randomly the reservoir volumes as particles for all the scheduling hours within the specified volume limits.
2. Find the global best for the first iteration and initialize the local best for each particle for the first iteration.
3. Start randomly the velocity vectors between the minimum values to the maximum values.
4. Start the main iteration loop.
5. Find the global best for each of the iterations.
6. Find and update the local best for each of the iterations.
7. For every of the iterations, update the particles' present locations and velocity using equation (9).
8. For each of the iterations, evaluate the fitness function.
9. Stop the program and print the results when the stopping criteria are reached.

3.2. Short Term Hydrothermal Scheduling Using Improved Particle Swarm Optimization(IPSO)

Reference [5] has astonished in using the phrase "Improved PSO" since it has not made any modification in the equation of the canonical PSO and used the same equation (9). The improvement is basically made in selecting the candidate for being the potential particle. It has used the discharge rate as a potential candidate of being a particle. Another astonishing observation is that the results' table is the same as are the results of [3]; however, the final cost achieved is mentioned very low as compared to the total cost obtained by Samudi, *et. al.*[3]. However, this might be a human error and we forget and forgive it to move towards the flow used by authors in reference [5] for the implementation, which is given below.

1. Start randomly the water discharge rate as particles for all the scheduling hours within the specified volume limits.
2. Find the global best for the first iteration and initialize the local best for each particle for the first iteration using the fitness function.
3. Start randomly the velocity vectors between the minimum values to the maximum values.
4. Start the main iteration loop.
5. Find the global best for each of the iterations.
6. Find and update the local best for each of the iterations.
7. For every of the iterations, update the particles'

present locations and velocities using equation (9).
8. For each of the iterations, evaluate the fitness function.
9. Stop the program and print the results when the stopping criteria are reached.

3.3. Short Term Hydrothermal Scheduling Using Fuzzy Adaptive PSO

Reference [6] improves the CPSO algorithm in such a way that the inertia weight *W* in equation (9) is improved in each of the iteration using a *fuzzyadaptive* system. According to authors in [6], it is the inertia weight that is used to enhance the search of global optimum solution. A large inertia weight helps in finding the global best while a small inertia weight makes the particles to roam about the local optimum.

A fuzzy system to let the inertia weight adapt works by taking two inputs; *the evolution speed factor (ESF) and the square deviation of the fitness (β).* If '*f*' is the cost function or the objective function of the optimization problem, then the evolution speed factor is calculated as;

$$ESF = \frac{G^k}{G^{k+1}} \tag{10}$$

Here,$G^k$is the global best particle till the k^{th}iteration. The square deviation of fitness tells the distribution of particles in search space and is given as;

$$\alpha = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\frac{f_i^k - f_{mean}^k}{\max\left(f_g^k - f_{mean}^k\right),\left(f_{mean}^k - f_b^k\right)}\right]^2} \tag{11}$$

Here, *i*, *mean*, *g*, *b* i.e. the subscripts are depicting the number of particles in the k^{th} iteration.

Using the two inputs, a table of rules for the fuzzy system is designed that then describes the error or difference in the value of inertia weight that is to be added in each value of weight of k^{th} iteration to give the weight for the new iteration.

$$W^{k+1} = W^k + \Delta W \tag{12}$$

The rules table for the fuzzy adaptive system is given in reference [6]. The flow points of Fuzzy adaptive PSO based short term hydrothermal scheduling are given as;

1. Start randomly the volume of water in the reservoir as particles for all the scheduling hours within the specified volume limits.
2. Find the global best for the first iteration and initialize the local best for each particle for the first iteration using the fitness function.
3. Start randomly the velocity vectors between the minimum values to the maximum values.
4. Start the main iteration loop.
5. Calculate ESF and the square deviation of the the fitness.
6. Modify the inertia weight using fuzzy rules and equation (12).
7. For each of the iterations, evaluate the fitness function.
8. Repeat from steps 2 to 7 till the stopping criteria is reached.
9. Stop the program and print the results.

3.4. Short Term Hydrothermal Scheduling Using Local and Global Vision of PSO

Reference [7] uses the two types of neighborhood topologies in canonical PSO and studies the impact of them on the searching ability of the solution to the objective function of the short term hydrothermal scheduling problem. Moreover it also discusses the two varieties in each of these topological modifications in terms of the use of either the inertia weight or the constriction factor. The four equations thus obtained that are applied in reference [7] are shown in the table below;

Table 1: Local and global vision of PSO

Version of PSO	Equations
GWPSO Global vision of PSO with inertia weight	$\begin{cases} V_j^{k+1} = W.V_j^k + c_1 rand(0,1)(P_{best} - x_j^k) + c_2 rand(0,1)(G_{best} - x_j^k) \\ x_j^{k+1} = x_j^k + V_j^{k+1} \end{cases}$ (13)
LWPSO Local vision of PSO with inertia weight	$\begin{cases} V_j^{k+1} = W.V_j^k + c_1 rand(0,1)(P_{best} - x_j^k) + c_2 rand(0,1)(L_{best} - x_j^k) \\ x_j^{k+1} = x_j^k + V_j^{k+1} \end{cases}$ (14)
GCPSO Global vision of PSO with constriction factor	$\begin{cases} V_j^{k+1} = K.\left[V_j^k + c_1 rand(0,1)(P_{best} - x_j^k) + c_2 rand(0,1)(G_{best} - x_j^k)\right] \\ x_j^{k+1} = x_j^k + V_j^{k+1} \end{cases}$ (15)
LCPSO Local vision of PSO with constriction factor	$\begin{cases} V_j^{k+1} = K.\left[V_j^k + c_1 rand(0,1)(P_{best} - x_j^k) + c_2 rand(0,1)(L_{best} - x_j^k)\right] \\ x_j^{k+1} = x_j^k + V_j^{k+1} \end{cases}$ (16)

In the above defined equations (13) to (16), $G_{best}$is the best particle among all the neighbors when the global vision of PSO is used. $L_{best}$is the best particle among all the neighbors of an individual particle *x* when local vision of PSO is used. By global vision it is meant that all the particles other than particle *x* are considered as neighbors of particle *x*,whereas, by local version, it is meant that each particle *x* has limited number of neighbors among all the neighbors, i.e. in reference [7], the number of local neighbors of each particle *x* are 2 which are the previous and the next residents of the particles' array.

W is the inertia weight varies from its maximum value to its minimum value linearly which is equal to;

$$W = W_{\max} - \left(\frac{W_{\max} - W_{\min}}{iteration_{\max}} \right) \times iteration_{present} \quad (17)$$

K is the constriction factor defined by reference [7] to be equal to 0.729. It is also known as the restriction factor since it helps in increasing the convergence behavior of PSO algorithm. According to reference [7] the local versions in both the cases have proved to be better in finding the optimal dispatch solution of the short term hydrothermal scheduling problem. The flow of this algorithm in short term hydrothermal scheduling problem is given as;

1. Start randomly the volume of water in the reservoir as particles for all the scheduling hours within the specified volume limits.
2. Find the global best for the first iteration and initialize the local best for each particle for the first iteration using the fitness function.
3. Start randomly the velocity vectors between the minimum values to the maximum values.
4. Start the main iteration loop.
5. Update the particles' positions and velocities using any of the four equations of Table 1.
6. For each of the iterations, evaluate the fitness function.
7. Repeat from steps 2 to 6 till the stopping criteria is reached.
8. Stop the program and print the results.

3.5. Short Term Hydrothermal Scheduling Using Self Organizing Hierarchical PSO (SPSO-TVAC)

Self-hierarchical PSO in reference [9] modifies the canonical PSO equation (6) by changing the acceleration coefficients c_1 and c_2. Since it is a time variant modification, so [9] has named it as self-organizing hierarchical particle swarm with time varying acceleration coefficients. At each of the iterations $i+1$, the acceleration coefficients are given as;

$$\begin{cases} c_{1,i+1} = c_{1i} - \left(\dfrac{c_{1,i} - c_{1,f}}{i_{\max}} \right) \times i_{present} \\ c_{2,i+1} = c_{2i} - \left(\dfrac{c_{2,i} - c_{2,f}}{i_{\max}} \right) \times i_{present} \end{cases} \quad (18)$$

SPSO-TVAC also uses the self re initializing process of generating or relocating the particles to new random positions if the particles start sticking to the local optima. The flow of this algorithm is;

1. Start randomly the volume of water in the reservoir as particles for all the scheduling hours within the specified volume limits.
2. Find the global best for the first iteration and initialize the local best for each particle for the first iteration using the fitness function.
3. Start randomly the velocity vectors between the minimum values to the maximum values.
4. Start the main iteration loop.
5. Modify the acceleration coefficients using (18) in each of the iterations.
6. For each of the iterations, evaluate the fitness function.
7. Repeat from step 1 to 6 if the particles are stuck to the local minimum locations, otherwise, repeat from steps 2 to 6 till the stopping criteria is reached.
8. Stop the program and print the results.

4. SHORT TERM HYDROTHERMAL SCHEDULING USING FULLY INFORMED PARTICLE SWARM OPTIMIZATION

Since the birth of Particle Swarm Optimization algorithm, many variants of it have been coming into existence for it is believed that the canonical version is sometimes not able to reach the global optima of the objective function. A new extension to the canonical PSO was recently introduced by the author of the first version, i.e. Kennedy along with Mendes and Neves in references [10-11]. The author has utilized FIPSO to solve the short term hydrothermal scheduling problem. It is observed that in classical PSO, each particle is not completely informed with its neighborhood. Therefore, according to the references [10] and [11], reaching towards the optima is not that effective as it can be when fully informed. However, in Fully Informed PSO, each individual particle is fully informed with its neighborhood. Its iterations are given as

$$\begin{cases} \vartheta_{i+1} = R\left(\vartheta_i + \dfrac{\sum_{n=1}^{N_i} Rand\left(0,\phi\right).(P_{nbr(n)} - X_i)}{N_i} \right) \\ X_{i+1} = X_i + \vartheta_{i+1} \end{cases} \quad (19)$$

Where ϑ is the velocity vector, R is the restriction coefficient which is considered 0.729 in our implementation, P_i is the best position found till the i^{th} iteration by the particle where X_i is the existing position of the particle, ϕ is considered equal to 4.1. $P_{nbr(n)}$ is the best position found by the neighbor till the i^{th} iteration by n^{th} neighbor. By fully informed, it is meant that for each of the iteration *i*, every particle has the following information:

- Its own position $\mathbf{X}_i$ at the end of each of the iteration
- the local best of each of its neighbor

Therefore, it is now possible for each of the particles to move the search space while getting influenced from all of the best possible locations found so far by each of its neighbor. And this is performed by every individual particle at the end of iteration.

In this research work, the system used to implement the FIPSO algorithm is that used by reference [1,3–5]. Volume of water in the reservoir is taken as the particles and the other variables like discharge rate, thermal power and hydal powerare taken as dependent variables.

The suggested FIPSO based algorithm implemented on short term hydrothermal scheduling problem has the following flow strategy;

1. Initialize the particles matrix, i.e. volume of water in the reservoir within the specified limits for each of the six scheduling periods.
2. Initialize randomly the velocity vectors. Velocity is defined within the maximum and minimum.

$$\begin{cases} V_{\max} = \dfrac{X_{\max} - X_{\min}}{no.\ of\ \ iterations} \\ V_{\min} = -(V_{\max}) \end{cases} \qquad (20)$$

Initialize randomly the vectors of local best for each of the particles. These vectors could be the same as the particles vectors for the first iteration but to make the program appearing more real, a separate matrix of local bests for each of the particles is initialized, using the same function as is used in the first step for generating the particles.

3. Produce the corresponding vectors of hydro-power, thermal power, discharge rate, individual cost and minimum cost.
4. If the constraints, hydro as well as thermal, as given in the problem statement are violated, set the particles within the limits.
5. For each of the iterations, find the fitness function using the particles, as well as the local bests. Compare the two results to update the vectors of local bests i.e. P_{nbr}.
6. Update the particles' locations using the FIPSO velocity update and particle update equation (19).
7. Go to the next iteration. Repeat the previous steps until the stopping criteria are reached.
8. Print the results.
9. Draw the graph between the global best solution of each of the iterations and the number of iterations. This graph will tell exactly the convergence behavior of the program.
10.

5. RESULTS OF SHORT TERM HYDROTHERMAL SCHEDULING USING FIPSO

The number of particles taken for implementing the FIPSO algorithm on short term hydrothermal scheduling problem is 15. The results are shown in Table 2 for the scheduling of hydro and thermal units. Each interval given in Table 2 is of 12 hours each. Since these algorithms are heuristic in nature, therefore, it is quite a difficult task to obtain the similar solution all the time. A lot of research is being done these days to improve the convergence characteristics of these heuristic algorithms. To overcome this problem, a statistical analysis between the maximum value obtained for the minimum cost and the minimum value obtained for the minimum cost is also given in Table 3. The convergence behavior of the global best solution of each of the iteration is also given in Figure1. Table 6 gives the comparison of proposed FIPSO based method with the two modifications of PSO of references [3-5]. The other modifications' results are not shown for comparison because they have not used the same system as used by [3-5].

Table 2: Results of short term hydrothermal scheduling using FIPSO method using 15 particles

Interval	Thermal Power (MW)	Hydro Power (MW)	Volume of water (acre-ft.)	Discharge rate (Acre-ft/hr)	Total cost of operation ($)
1	858.1	341.9	99649	2029.2	669650
2	505.3	994.7	60365	5273.6	
3	1100	0	99779	0	
4	1094.5	705.5	77746	3836.1	
5	773.6	176.4	87263	1206.9	
6	506.9	793.1	60000	4271.9	

Table 2: Statistical analysis of the proposed method

Minimum cost	Maximum cost	Average cost
669650	713680	691665

Table 3: Comparison among proposed algorithm and previously implemented two forms of PSO

Works	Minimum cost
Samudai*et al* [3]	693428.5
Padamini*et al*[5]	693426.2
Sinha *et al* [4]	709862.048
Proposed FIPSO	669650.0

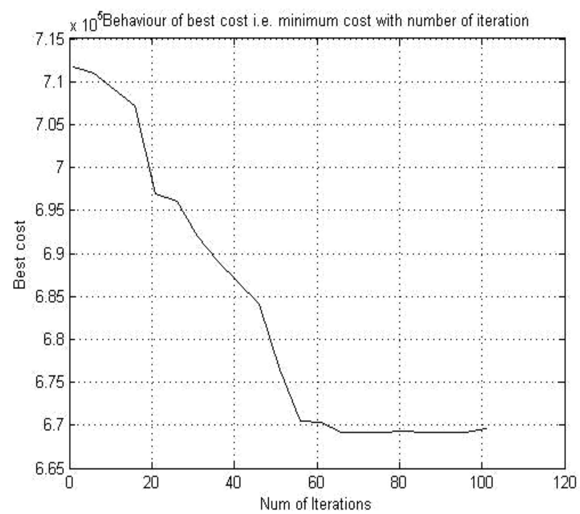

Figure1: Convergence behavior of the cost obtained by inserting the global best particle of each of the iterations

6. CONCLUSION

Short term hydrothermal scheduling problem which is a kind of economic dispatch problem has been solved using many conventional and heuristic techniques of optimization. Particle swarm optimization and its variants have performed the best among all. In this paper, different modifications in CPSO applied till date on the short term hydrothermal scheduling problem has been discussed. A reader can find helping material in this review to learn all these implementations. Moreover, a new variant of CPSO known as fully informed particle swarm optimization (FIPSO) is also implemented according to the steps described. It is interesting to know the fact that FIPSO has outperformed all the previous versions of CPSO for the same problem of interest.

REFERENCES

[1] A. J. Wood and B. F. Wollenberg, Power generation, operation, and control: John Wiley & Sons, 2012.

[2] S. Sivanagaraju, Power system operation and control: Pearson Education India, 2009.

[3] C. Samudi, G. Das, P. Ojha, T. Sreeni, and S. Cherian, "Hydro thermal scheduling using particle swarm optimization",*Proceeding of Transmission and Distribution Conference and Exposition*, Chicago (USA), pp. 1-5.21-24 April, 2008.

[4] N. Sinha and L.-L. Lai, "Meta heuristic search algorithms for short-term hydrothermal scheduling",*Proceedeings of International Conference on, Machine Learning and Cybernetics,* , Dalian (China), pp. 4050-4056,13-16 August, 2006

[5] S. Padmini and C. Rajan, "Improved PSO for short term hydrothermal scheduling",*Proceedings ofInternational Conference on Sustainable Energy and Intelligent Systems*, Chennai (India), pp. 332-334.20-22 July,2011.

[6] W. Chang, "Optimal Scheduling of Hydrothermal System Based on Improved Particle Swarm Optimization",*Proceedings of Power and Energy Engineering Conference (APPEEC)*, Asia-Pacific, pp. 1-4,28-31 March,2010.

[7] B. Yu, X. Yuan, and J. Wang,"Short-term hydro-thermal scheduling using particle swarm optimizationmethod."*Energy Conversion and Management*, **48**(7), pp. 1902-1908.2007.

[8] S. Thakur, C. Boonchay, and W. Ongsakul, "Optimal hydrothermal generation scheduling using self-organizing hierarchical PSO",*Proceedings of IEEE Power and Energy Society General Meeting, Minneapolis*, MN, pp. 1-6.25-29 July, 2010.

[9] M. R. Alrashidi and E. El-Hawary, “A survey of Particle Swarm Optimization in applications in power system operations”, *Electric Power Component and Systems*, **Vol. 34**, pp. 1349-1357,12 December, 2006.

[10] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better", *IEEE Transactions onEvolutionary Computation*,**vol. 8**, pp. 204-210, 2004.

[11] J. Kennedy and R. Mendes, "Neighborhood topologies in fully informed and best-of-neighborhood particle swarms",*IEEE Transactions on Systems Man and Cybernetics Part C Applications and Reviews*, **vol. 36**, p. 515, 2006.