International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

Cooperative Multiple Particle Swarm Optimization (CMPSO) and Spatial Extended Particle Swarm Optimization (SEPSO) For Solving Reactive Power Optimization Problem

K. Lenin, B. Ravindranath Reddy and M. Surya Kalavathi

Electrical and Electronics Engineering, Jawaharlal Nehru Technological University, Kukatpally, Hyderabad, India

*Corresponding Author's E-mail: *gklenin@gmail.com*

Abstract

Reactive Power Optimization is a complex combinatorial optimization problem involving non-linear function having multiple local minima, non-linear and discontinuous constrains. This paper presents Cooperative Multiple Particle Swarm Optimization (CMPSO) and Spatial Extended Particle Swarm Optimization (SEPSO) in trying to overcome the Problem of premature convergence. CMPSO and SEPSO are applied to Reactive Power Optimization problem and are evaluated on standard IEEE 30Bus System. The results show that CMPSO prevents premature convergence to high degree but still keeps a rapid convergence. It gives best solution when compared to Spatial Extended Particle Swarm Optimization (SEPSO).

Keywords: spatial extended, particle Swarm, cooperative multiple, Reactive Power Optimization.

1. Introduction

The reactive power optimization problem has a significant influence on secure and economic operation of power systems. The reactive power generation, although itself having no production cost, does however affect the overall generation cost by the way of the transmission loss. A procedure, which allocates the reactive power generation so as to minimize the transmission loss, will consequently result on the lowest production cost for

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

which the operation constraints are satisfied. The operation constraints may include reactive power optimization problem. The conventional gradient-based optimization algorithm has been widely used to solve this problem for decades. Obviously, this problem is in nature a global optimization problem, which may have several local minima, and the conventional optimization methods easily lead to local optimum. On the other hand, in the conventional optimization algorithms, many mathematical assumptions, such as analytic and differential properties of the objective functions and unique minima existing in problem domains, have to be given to simplify the problem. Otherwise it is very difficult to calculate the gradient variables in the conventional methods. Further, in practical power system operation, the data acquired by the SCADA (Supervisory Control and Data Acquisition) system are contaminated by noise. Such data may cause difficulties in computation of gradients. Consequently, the optimization could not be carried out in many occasions. In the last decade, many new stochastic search methods have been developed for the global optimization problems such as simulated annealing, genetic algorithms and evolutionary programming.

2. PROBLEM FORMUMATION

The objective of the reactive power optimization problem is to minimize the active power loss in the transmission network as well as to improve the voltage profile of the system. Adjusting reactive power controllers like generator bus voltages, reactive power of VAR sources and transformer taps performs reactive power scheduling.

$$\min P_L = \sum_{i=1}^{NB} P_i(X, Y, \delta) \quad (1)$$

Subject to

i) The control vector constraints

$$X_{min} \leq X \leq X_{max} \quad (2)$$

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

ii) The dependent vector constraints

$$Y_{min} \leq Y \leq Y_{max} \quad (3)$$

and

iii) The power flow constraint

$$F(X, Y, \delta) = 0 \quad (4)$$

where

$$X = [V_G, T, Q_C] \quad (5)$$

$$Y = [Qg, V_L, I] \quad (6)$$

NB	-	number of buses in the system.
δ	-	vector of bus phase angles
P^i	-	real power injection into the i^{th} bus
V_G	-	vector of generator voltage magnitudes
T	-	vector of tap settings of on load transformer tap changer.
Q_C	-	vector of reactive power of switchable VAR sources.
V_L	-	vector of load bus voltage magnitude.
I	-	vector of current in the lines.
P_L	-	vector of power loss in the transmission network.

3. PARTICLE SWARM OPTIMIZATION (PSO)

The inclination of the researchers towards the implementation of the biologically inspired algorithms in solving the engineering problems have led to the invention of many algorithms such as Genetic Algorithm, Ant colony optimization, Artificial Immune System based algorithms etc. Particle swarm optimization (PSO) is one of the biologically inspired evolutionary algorithms which drive the idea from the flocking of

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

birds. Abundant examples could be extracted from the nature that demonstrates that social sharing of information among the individuals of a population may provide an evolutionary advantage. PSO was first proposed by Kennedy and Eberhart (1995) and it has been deserved considerable attention in recent years in the global optimization areas. PSO originally intends to graphically mimic the elegant way in which swarms find their food sources and save themselves from predators (Eberhart and Kennedy 1995). It is a population-based stochastic optimization paradigm, in which each individual termed as particle from the population of swarm changes their position with time and represent a potential solution.

PSO in some ways resembles with the other existing Evolutionary Algorithms, such as Genetic Algorithm, but the difference lies in its definition in a social context rather than biological context. According to Eberhart and Shi (2001) PSO is based on simple concepts with the ease of implementation and computational efficacy. Particle Swarm Optimization (PSO) algorithm motivated by the flocking of the birds works on the social behavioral interaction among the particles in the swarm. It begins with the random initialization of a population of particles in the search space. These particles are considered to be in multidimensional space (D-dimensional) where each particle has a position and velocity. These two factors i.e. the position and velocity demonstrates the particles status in the search space.

Hence in a PSO system, particles fly/move around in multi directions in the search space, and the position of each particle is guided accordingly by the memory of their own best position, as well as of a neighboring particle. These particles communicate the best positions to each other and adjust their own position and velocity accordingly. Parsopoulos and Vrahatis (2002) proposed basically two main variants of the PSO algorithm:

• Global neighborhood, where best global position is communicated to all particles and updated immediately in the swarm.

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

- Local neighborhood, where each particle moves towards its best previous position and towards the best particle in its restricted neighborhood. Towards the best particle in its restricted neighborhood.

In the proposed work the global variant has been adapted. The reason behind opting for the

global neighborhood is due to the fact that local neighborhood even though allows better

exploration of the search space and reduces the susceptibility of PSO to falling into local

minima; it slows down the convergence speed. The position and velocity vectors of the ith

particle can be represented as

xi = (xi1, xi2, . . . , xiD) (7)

pi = (pi1, pi2, . . . , piD) (8)

The fittest particle among all the particles in the population is represented by

F = (f1, f2, fD) (9)

The velocity vector for the ith particle can be represented as

Vi = (vi1, vi2, …viD) (10)

The updated velocity and position for the next fitness evaluation of each particle could be determined according to the following equations:

$$v_{id}^{k+1} = \omega \cdot v_{id}^{k} + c_1 \cdot R_1() \cdot \left(p_{id}^{k} - x_{id}^{k}\right) + c_2 \cdot R_2() \cdot (f_d^k - x_{id}^{k}) \quad (11)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (12)$$

Here k is the iteration number, d = 1, 2. . . D; i = 1, 2. . . N, and N is the size of the population (swarm). c1 and c2 are two positive values called acceleration constants, R1( )

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

and R2( ) are two independent random numbers that uniformly distribute between 0 and 1 and are used to stochastically vary the relative pull of pi and f (Clerc and Kennedy 2002). The introduction of such random elements into the optimization is intended to simulate the slightly unpredictable component of natural swarm behaviour. ω is the inertial weight introduced by Shi and Eberhart (1998b) in order to improve the performance of the particle swarm optimizer. The equation (12) contains the three terms on the right hand side in which the inertial effects of the movement is represented by the first term.

The memory of the individual and whole is referred by second and third terms respectively. Basically, equation (12) is used to calculate the particle's new velocity which depends on its preceding velocity and the distances of its present position from both its own best past position and the group's best past position. All the other particles follow the best position found and moves closer to it, exploring the region more thoroughly in the process. According to Robinson and Rahmat-Samii (2004) equation (12) is the central constituent of the entire optimization. The stochastic tendency to return towards particle's preceding best position is represented by the second term of the equation (12). The third term in equation (12) is referred as social influence term. The variable F keeps moving towards the best solution i.e. optimal solution found by the neighbour particles in the search space.

Particle farther from the global best is more strongly pulled from its location, and moves rapidly than a closer particle. The particles velocity comes to halt after it reaches the locations of best fitness. This is the point from where the particles are pulled back in the opposite direction. The performances of the individual particles are evaluated by a predefined fitness function dependent on the problem during the evolution of the swarm. In case of maximization of the fitness function Fitness (xi), the individual best position of each particle pi and the global best position f are updated after each iteration using the following two equations, respectively:

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

$$p_i^{k+1} = \begin{cases} x_i^{k+1} : Fit\,(x_i^{k+1}) > Fit(p_i^{k+1}) \\ p_i^k : Fit(x_i^{k+1}) \leq Fit(p_i^{k+1}) \end{cases} \tag{13}$$

$$p_i^k : Fit(x_i^{k+1}) \leq Fit(p_i^{k+1}) \tag{14}$$

$$F^{k+1} = \arg\max Fit\,(p_i^{k+1}) \tag{15}$$

Where 'Fit' refers to the fitness value for the respective iteration.

4. SPATIAL EXTENDED SPATIAL EXTENDED PARTICLE SWARM OPTIMIZATION (SEPSO)

The main motivation for giving the particles an extension in space is that the particles in the basic PSO tend to cluster too closely [8]. When an optimum (local or global) is found by one particle the other particles will be drawn towards it. If all particles end up in this optimum, they will stay at this optimum without much chance to escape. This simply happens because of the way the basic PSO (in particular the velocity update formula) works. If the identified optimum is only local it would be advantageous to let some of the particles explore other areas of the search space while the remaining particles stay at this optimum to fine tune the solution [9].

In our spatial particle extension model, we tried to increase the diversity when particles started to cluster. For this we added a radius r to each particle in order to check whether two particles would collide. If they collide, action can be taken to make them bounce off to avoid the collision and thus the clustering. An important issue is to determine the direction in which the particles should bounce away and at what speed. We investigated three strategies: 1) Random bouncing, where the particles are sent away from the collision in a random direction preserving the old speed. 2) Realistic physical bouncing. 3) Simple velocity-line bouncing in which the particles continue to move in the direction of their old velocity-vector, but with a scaled speed. This gives the particle the

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

possibility of making an U-turn and return to where it came from (by scaling with a negative bounce-factor) [10]. Particles can be slowed down (bounce-factor between 0 and 1) or speeded up to avoid the collision (bounce-factor greater than 1). A particle collision avoidance with velocity-line bouncing is illustrated in figure 1.

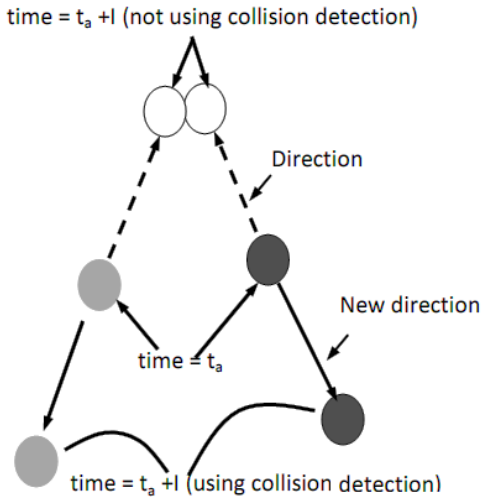

Figure 1: A particle-collision. The solid circles are the particle positions at time t_0 and t_{0+1} using collision detection. The dotted rings are their positions at time t_{0+1} without collision detection. The bounce-factor is set to -1.

A. ALGORITHM OF RPO USING SEPSO

The proposed RPO algorithm using the SEPSO can be expressed as follows:

Step 1, Initial searching points and velocities of agents are generated using the above-mentioned state variables randomly.

Step 2 Ploss to the searching points for each agent is calculated using the load flow calculation. If the constraints are violated, the penalty is added to the loss (evaluation value of agent).

The fitness function of each particle is calculated as:

$$f_n = P^n{}_L + \alpha \sum_{j=1}^{NG} Q_{G,j}^{\lim,n} + \beta \sum_{j=1}^{NL} V_{L,j}^{\lim,n} \quad ; n = 1,2..,N_n \qquad (16)$$

α, β = penalty factors

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

P^n_L = total real power losses of the n^{-th} particle

$$Q^{lim,n}_{G,j} = \begin{cases} Q_{G,min} - Q^n_{G,j} & \text{if } Q^n_{G,j} < Q_{G,min} \\ Q^n_{G,j} - Q_{G,max} & \text{if } Q^n_{G,j} > Q_{G,max} \end{cases} \quad (17)$$

and

$$V^{lim,n}_{L,j} = \begin{cases} \left|V^n_{L,j}\right| - V_{L,max}, & \text{if } \left|V^n_{L,j}\right| > V_{L,max} \\ 0 & \text{otherwise} \end{cases} \quad (18)$$

Step 3. Pbest is set to each initial searching point. The initial best evaluated value (loss with penalty) among pbests is set to gbest.

Step 4. New velocities are calculated using (12). The continuous equations are utilized for continuous variables and the discrete equations for discrete variables.

Step 5. New searching points are calculated using (13). The continuous equations are utilized for continuous variables and the discrete equations for discrete variables.

Step 6. Ploss to the new searching points and the evaluation values are calculated.

Step 7. Radius r is allotted to each particle in order to check whether two particles collide and if collusion means bounce factor added to avoid collusion.

Step 8. If the evaluation value of each agent is better than the previous pbest, the value is set to pbest. If the best pbest is better than gbest, the value is set to gbest. All of gbests are stored as candidates for the final control strategy.

Step 9. If the iteration number reaches the maximum iteration number, then stop. Otherwise, go to Step 4. If the voltage and power flow constraints are violated, the absolute violated value from the maximum and minimum

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

boundaries is largely weighted and added to the objective function as a penalty term.

5. Cooperative Multiple Particle Swarm Optimization (CMPSO) algorithm & its implementation

This section describes the formulations of the CMPSO algorithm for the process planning and scheduling in multi plant supply chain scenario. The prime objective of the problem considered is to generate an operation sequence and simultaneously select an appropriate machine corresponding to each operation from existing alternatives. It is a multiple dimensional problem as shown in the Figure 4. In the figure, first row represents an operation sequence while the second row represents the machine corresponding to each operation. In order to resolve the complexity of the problem in this piece of research CMPSO algorithm has been proposed. One of the key issues in successful implementation of PSO to a specified engineering problem is the representation scheme, i.e. finding a suitable mapping between the problem solution and the PSO particle. In the proposed methodology during the exploration of the search space the sister swarms cooperate with each other.

In this paper each bit of solution are positive integers and comprises a non-continuous integer search space. Since the original PSO works on a real-valued search space, especially on the particle positions (i.e., the operation sequence and corresponding machine in this paper) is calculated using equation (13) which is real numbers. Hence, a conversion is needed between the real-valued positions and the positive-integer-valued indices. In order to meet the criteria the sign is ignored and value is changed to the closest integer. After calculating the x_{id}^{k+1} term in equation (13) the changes mentioned earlier are applied leaving the rest part of the equations (12) and (13) same as in the original PSO. These changes do not have any effect on the performance of the algorithm and has been proven to be feasible (Salman et al. 2002, Laskari et al. 2002, Parsopoulos and Vrahatis

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

2002). The individuals in the swarm are initialized by randomly setting their positions and velocities using operation sequence or machine depending on the nature of the swarm. During the iteration, reset is performed only when the value of a new position component calculated from equation (13) is greater than the upper limitation of the search space. It should be noted that, in the first row at any time of the operation process two bits cannot have the same values. Hence, if a new component value is calculated using equation (13) (for first row) that already exists, a random small integer will be added to this value till no collision exists. This combination pfacilitates fast convergence and ensures near-optimal solutions by establishing a proper balance between exploration and exploitation.

In case of simple PSO in equation (12) random numbers were generated using the Random function. However, during experimentation it has been found that random functions are associated with some demerits. Hence, in order to overcome the demerits of the random number in this research not only it is being replaced by chaotic sequences, but also a new hybrid chaotic sequence has been proposed. This paragraph explains the significance of applying a chaotic sequence generator to update the velocity instead of the random number generator. The random function used in equation (12) has been replaced with a chaotic function because of the ergodic and stochastic properties of the chaotic systems.

One of the limitations coupled with the random number generators is that the solution becomes conserved by sequential correlation of successive cells; hence requiring more number of generations to converge towards an optimal or near-optimal solution. Also the commonly used random number generators have a tendency to generate the higher-order part more randomly than their lower-order counterpart (Caponetto et al., 2003). Therefore, it requires a consistent random number generator which can explore search space without being biased. Recently, various chaotic sequences have been applied in areas related to secure transmission, neural networks, natural phenomena modeling,

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

deoxyribonucleic acid computing procedures, and non-linear circuits (Arena et al., (2000), Determan and Foster (1999), Manganaro and Pineda (1997), Sugantahn (1999), Nozawa (1992), and Wang and Smith (1998)) and encouraging results have been obtained with random number generators. The unpredictability characteristics, i.e. spread spectrum characteristics, justify theoretically the use of a chaotic sequence. Thus, the recent research drift towards the implementation of chaotic sequence generators in various AI tools motivated us to use in the present problem scenario.

The commonly used chaotic equations by researchers are Logistic map-based chaos equation (LM), Tent map-based chaos equation (TM), Sinusoidal integrator-based chaos equation (SI), and Gauss map-based chaos equation (GM). As usual each equation has some advantage and some disadvantage and in order to overcome the demerit of each chaotic equation in this present research a hybrid chaotic equation termed as Chaotic Sequence-based Hybrid chaos equation (HC) has been proposed. The proposed chaotic equation incorporates the advantages of each chaotic equations mentioned below;

(A) Logistic map-based chaos equation (LM): In this method logistic map-based chaotic Sequence is used to generate random numbers. It is one of the simplest dynamic systems evidencing chaotic behavior. The logistic map chaotic equation is delineated as follows.

$$Y_{k+1} = \omega\, Y_k(1 - Y_k)$$

Where ω is tuning parameter

(B) Tent map-based chaos equation (TM): In this method, random numbers are generated using Tent map-based chaotic sequence. It resemble as the logistic map which follows the following equations.

$$Y_{k+1} = \mu_k\,(Y_k) \qquad (19)$$

$$p_i^{k+1} = \begin{cases} \frac{Y_k}{0.7}, & if\ Y_k < 0.7 \\ \frac{1}{0.3} Y_k(1 - Y_k), & otherwise \end{cases} \qquad (20)$$

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

(C) Sinusoidal integrator-based chaos equation (SI): In this chaotic equation, random Numbers are generated using the following Sinusoidal Integrator relation:

$$Y_{k+1} = \sin(\pi\, Y_k) \tag{21}$$

$$Y_{k+1} = \mu_k\,(Y_k) \tag{22}$$

$$\mu(Y_k) = \begin{cases} 0, & if\ Y_k = 0 \\ \frac{1}{Y_k} mod1, & Y_k \in (0,1) \end{cases} \tag{23}$$

(D) Gauss map-based chaotic equation (GM): In this chaotic equation, Gauss Map function is used to generate the random numbers and it transfers from one stage to another stage in a quadratic manner. Gauss Map function can be expressed as follows:

(E) Chaotic sequence-based hybrid chaotic equation (HC): Randomly select a chaotic sequence strategy among aforementioned four strategies and generate random number using selected chaotic equation.

As mentioned earlier first row of each solution represents the operation sequence and second row represents the corresponding machine. For each individual row the proposed CMPSO algorithm runs separately. After updating the position and velocity for each row the sister swarm will cooperate with each other to evaluate the fitness function. On the basis of the computed fitness value the global and local best positions are decided. And after certain number of iterations the solution will tend to converge towards the optimality or sub-optimality. The steps of the proposed CMPSO algorithm for the reactive power dispatch problem are shown below:

Step 1: Generate discrete search space for first and second row of solution i.e. maximum numbers of operation and number of possible machine corresponding to each option.

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

Step 2: Generate random initial solution, and assign random position X and velocity V vectors corresponding to each particle swarm (For both sister swarms) and assign number of generation (num_gen=1)

Step 3: Calculate the fitness value by the help of sister swarm and update the personal best Position and global best position of each of sister swarm using equation (14) and (15) respectively.

Step 4: Update the velocity and position of the each swarm using the chaotic sequence mentioned above.

Step 5: num_gen=num_gen+1;

Step 6: If num_gen=max num_gen; go to step 7 otherwise go to step 3

Step7: Terminate the algorithm and global position is the optimal solution obtained by the algorithm.

6. Simulation results

The proposed CMPSO and SEPSO algorithm for RPO problem has been evaluated on Standard IEEE 30 bus system.

NB = 30, NL = 41, NG = 6, NTR = 4 , Population size = 50, V_{min} - 0.95 ,V_{max} - 1.05
T_{min} - 0.9 ,T_{max} -1.1 ,sus_{max} - 0.15 ,sus_{min} - 0.0

Table 1: Optimum Reactive Power Schedule Values Obtained For IEEE 30 Bus Systems.

	SEPSO	**CMPSO**
No. of Iteration	188	162
Population size	50	50
Time (sec.)	9.317	7.998
Loss (MW)	9.4797	9.3942

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

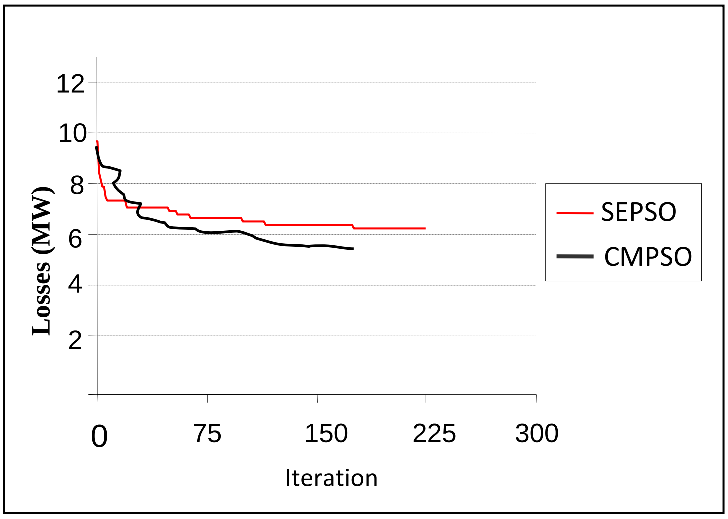

Figure 2: Convergence Characteristics of IEEE 30 Bus System

Table 2: Optimal Control Values of Standard IEEE 30 Bus System

VG1	VG2	VG3	VG4	VG5	VG6	T1	T2	T3	T4
1.05	1.03	1.01	1.01	1.07	1.09	0.99	1.95	1.00	0.94

Conclusion

In this paper CMPSO and SEPSO algorithm has been developed for determination of global optimum solution for reactive power optimization problem. The performance of the proposed algorithm demonstrated through its evaluation on IEEE 30 bus power system shows that CMPSO is able to undertake global search with a fast converges rate and a future of robust computation. From the simulation study it has been found that CMPSO converges to the global optimum than SEPSO.

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

References

[1]. K.Y. Lee, Y.M. Park and J.L. Ortiz "A united approach to optimal real and reactive power dispatch" IEEE, Trans. Power System, vol.1 No.2, pp.1147 - 1153, May 1985.

[2]. Y.Y. Hong, D.I. Sun, S.Y. Lin, C.J. "Multiyear Multi case optimal AVR planning" IEEE, Trans. Power System, (1990), vol.5 No.4, pp.1294 – 1301.

[3]. N. Deeb, S.M. Shaidepour "Linear reactive power optimization in large network using the decomposition approach", IEEE, Trans. Power System, vol.5 No.2, pp.428 - 435, May 1990.

[4]. S. J. Vesterstrom and J. Riget, "A Diversity-Guided Particle Swarm Optimizer the ARPSO", EVALife Technical Report No. 2002.

[5]. M A. Abido, "Optimal power flow using particle swarm optimization", Electric Power Energy Systems, (2002), Vol.26, Pages 563–571.

[6.]. R. Caponetto, L. Fortuna, S. Fazzino and, M. G. Xibilia "Chaotic sequences to improve the performance of evolutionary algorithm". IEEE Trans' (2003), Vol.7, No.3, Pages 289–304.

[7]. W. Cedeño, and K. Dimitris "Using particle swarms for the development of QSAR models based on K-nearest neighbour and kernel regression", Journal of Computer-Aided Molecular Design, (2003), Vol.17, No.2-4, Pages 255-263, ISSN0920-654X (Print) 1573-4951 (Online).

[8]. K. Thiemo, S. J. Vesterstrom and R. Jacques, "Particle Swarm Optimization with Spatial Particle Extension", Procs. of CEC_2002.

[9]. Y. Fukuyama, "Practical Distribution State Estimation Using Hybrid Particle Swarm Optimization", Proc. of IEEE Power Engineering Society Winter Meeting, Columbus, 2001.

[10]. R. C. Eberhart and Y. Shi "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization", Proc. of CEC 2000.

[11].Y. Fukuyama "A Particle Swarm Optimization for Reactive Power and voltage control in Electric Power System", IEEE Trans. on Power Systems, pp. 87 – 93, 2001.

[12]. H.W. Dommel, W.F. Tinney, "Optimal power flow solutions", IEEE PAS, vol. 87, Oct 1968, pp. 1866-1876.

[13]. O.J. Bright, M. Prais, B. Stott, "Further developments in LP-based OPF", IEEE Trans. on Power Systems, vol. 5, no. 3, Aug 1990, pp. 697-711.

[14]. B. Barán, J. Vallejos, R. Ramos and U. Fernández, "Reactive Power Compensation using a Multi-Objective Evolutionary Algorithm" in Proc. IEEE Porto PowerTech'2001. Porto, Portugal. 2001.

[15]. J. Vallejos, R. Ramos and B. Barán, "Multi-Objective Optimization in Reactive Power Compensation" Jornadas de Informática y Telecomunicaciones - Conferencia de Informáticay Tecnología Aplicada (JITCITA 2001). Asuncion, Paraguay. 2001.

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(9), Oct, 2013, pp. 345-361, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

- -

[16]. C.A. Coello and M.S. Lechuga, "MOPSO: A proposal for multiple objective particle swarm optimization. In Proc. 2002 IEEE Congress on Evolutionary Computation, pages 1051–1056, Hawaii, HI, USA, 2002.

Authors

K. Lenin has received his B.E., Degree, electrical and electronics engineering in 1999 from university of madras, Chennai, India and M.E., Degree in power systems in 2000 from Annamalai University, TamilNadu, India. Working as Assistant professor in sree sastha college of engineering & at presently pursuing Ph.D., degree at JNTU, Hyderabad,India.

Bhumanapally. RavindhranathReddy, Born on 3rd September,1969. Got his B.Tech in Electrical & Electronics Engineering from the J.N.T.U. College of Engg., Anantapur in the year 1991. Completed his M.Tech in Energy Systems in IPGSR of J.N.T.University Hyderabad in the year 1997. Obtained his doctoral degree from JNTUA,Anantapur University in the field of Electrical Power Systems. Published 12 Research Papers and presently guiding 6 Ph.D. Scholars. He was specialized in Power Systems, High Voltage Engineering and Control Systems. His research interests include Simulation studies on Transients of different power system equipment.

M. Surya Kalavathi has received her B.Tech. Electrical and Electronics Engineering from SVU, Andhra Pradesh, India and M.Tech, power system operation and control from SVU, Andhra Pradesh, India. she received her Phd. Degree from JNTU, hyderabad and Post doc. From CMU – USA. Currently she is Professor and Head of the electrical and electronics engineering department in JNTU, Hyderabad, India and she has Published 16 Research Papers and presently guiding 5 Ph.D. Scholars. She has specialised in Power Systems, High Voltage Engineering and Control Systems. Her research interests include Simulation studies on Transients of different power system equipment. She has 18 years of experience. She has invited for various lectures in institutes.