[Lenin *, Vol.6 (Iss.4): April 2018] ISSN- 2350-0530(O), ISSN- 2394-3629(P)
(Received: Mar 09, 2018 - Accepted: Apr 26, 2018) DOI: 10.5281/zenodo.1248180

INTERNATIONAL JOURNAL OF RESEARCH – GRANTHAALAYAH
A knowledge Repository

Science

DIMENSIONED PARTICLE SWARM OPTIMIZATION FOR REACTIVE POWER OPTIMIZATION PROBLEM

Dr.K. Lenin [*1]
[*1] Professor, Department of EEE, Prasad V. Potluri Siddhartha Institute of Technology, Kanuru, Vijayawada, Andhra Pradesh -520007, India

Abstract

This paper present's Dimensioned Particle Swarm Optimization (DPSO) algorithm for solving Reactive power optimization (RPO) problem. Dimensioned extension is introduced to particles in the particle swarm optimization (PSO) model in order to overcome premature convergence in interactive optimization. In the performance of basic PSO often flattens out with a loss of diversity in the search space as resulting in local optimal solution. Proposed algorithm has been tested in standard IEEE 57 test bus system and compared to other standard algorithms. Simulation results reveal about the best performance of the proposed algorithm in reducing the real power loss and voltage profiles are within the limits.

Keywords: Particle Swarm Optimization; Real Power Loss; Spatial.

Cite This Article: Dr.K. Lenin. (2018). "DIMENSIONED PARTICLE SWARM OPTIMIZATION FOR REACTIVE POWER OPTIMIZATION PROBLEM." *International Journal of Research - Granthaalayah*, 6(4), 281-290. https://doi.org/10.5281/zenodo.1248180.

1. Introduction

Reactive power optimization (RPO) problem is one of the difficult optimization problems in power systems. The sources of the reactive power are the generators, synchronous condensers, capacitors, static compensators and tap changing transformers. The problem that has to be solved in a reactive power optimization is to determine the required reactive generation at various locations so as to optimize the objective function. Here the reactive power dispatch problem involves best utilization of the existing generator bus voltage magnitudes, transformer tap setting and the output of reactive power sources so as to minimize the loss and to enhance the voltage stability of the system. It involves a non linear optimization problem. Various mathematical techniques have been adopted to solve this optimal reactive power dispatch problem. These include the gradient method [1-2], Newton method [3] and linear programming [4-7].The gradient and Newton methods suffer from the difficulty in handling inequality constraints. To apply linear programming, the input- output function is to be expressed as a set of linear functions which may lead to loss of accuracy. Recently Global Optimization techniques such as genetic algorithms have been proposed to solve the reactive power flow problem [8, 9]. This paper present Dimensioned Particle Swarm Optimization

(Received: Mar 09, 2018 - Accepted: Apr 26, 2018) DOI: 10.5281/zenodo.1248180

(DPSO) for solving Reactive power optimization problem. Dimensioned extension is introduced to particles in the particle swarm optimization (PSO) model in order to overcome premature convergence in interactive optimization. In the performance of basic PSO [10-13] often flattens out with a loss of diversity in the search space as resulting in local optimal solution. Proposed algorithm has been tested in standard IEEE 57 test bus system and compared to other standard algorithms. Simulation results reveal about the best performance of the proposed algorithm in reducing the real power loss and voltage profiles are within the limits.

2. Problem Formulation

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.

$$\text{Min } P_L = \sum_{i=1}^{NB} P_i(X, Y, \delta) \qquad \ldots \qquad (1)$$

Subject to

i) The control vector constraints

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

ii) The dependent vector constraints

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

And

iii) The power flow constraint

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

Where

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

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

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. Basic Particle Swarm Optimization (PSO) Model

Particle Swarm Optimization (PSO) has been developed through simulation of simplified social models. The features of the method are as follows:

1) The method is based on researches about swarms such as fish schooling and a flock of birds.
2) It is based on a simple concept. Therefore, the computation time is short and it requires few memories.

According to the research results for a flock of birds, birds find food by flocking (not by each individual). The observation leads the assumption that every information is shared inside flocking. Moreover, according to observation of behavior of human groups, behavior of each individual (agent) is also based on behavior patterns authorized by the groups such as customs and other behavior patterns according to the experiences by each individual. The assumption is a basic concept of PSO. PSO is basically developed through simulation of a flock of birds in two-dimension space. The position of each agent is represented by XY-axis position and the velocity (displacement vector) is expressed by vx (the velocity of X-axis) and vy (the velocity of Y-axis). Modification of the agent position is realized by using the position and the velocity information.

Searching Procedure of PSO

Searching procedures by PSO based on the above concept can be described as follows: a flock of agents optimizes a certain objective function. Each agent knows its best value so far (pbest) and its XY position. Moreover, each agent knows the best value in the group (gbest) among pbests, namely the best value so far of the group. Figure 1 shows the above concept of modification of searching points. The modified velocity of each agent can be calculated using the current velocity and the distance from pbest and gbest as shown below:

$$v^{k+1} = w_i v_i^{k} + c_1\, rand \ \text{x}\ (pbest_i, - s_i^{k} + c_2 rand \ \text{x}\ (gbest - s_i^{k}) \ \ldots \qquad (7)$$

Where,
V_i^k : current velocity of agent i at iteration k,
V_i^{k+1} : modified velocity of agent i,
rand : random number between 0 and 1,
s_i^k : current position of agent i at iteration k,
$pbest_i$: pbest of agent i,
gbest : gbest of the group,
W_i : weight function for velocity of agent i,
C_i : weight coefficients for each term.

Using the above equation, a certain velocity that gradually gets close to pbests and gbest can be calculated. The current position (searching point in the solution space) can be modified by the following equation:

$$S_i^{k+1} = s_i^k + V_i^{k+1} \quad \dots \quad (8)$$

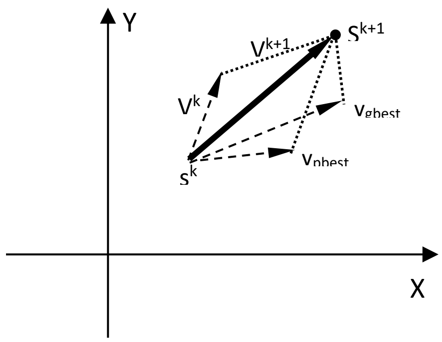


Figure 1: shows the above concept of modification of searching points

S^k : current searching point,
S^{k+1} : modified searching point,
V_k : current velocity,
V^{k+1} : modified velocity,
V_{pbest} : velocity based on pbest
V_{gbest} : velocity based on gbest

4. PSO Model with Dimensioned Particle Extension

The main motivation for giving the particles an extension in space is that the particles in the basic PSO tend to cluster too closely. 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 . 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 possibility of making an U-turn and return to where it came from (by scaling with a negative bounce-factor) . 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 2.

 ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: 10.5281/zenodo.1248180

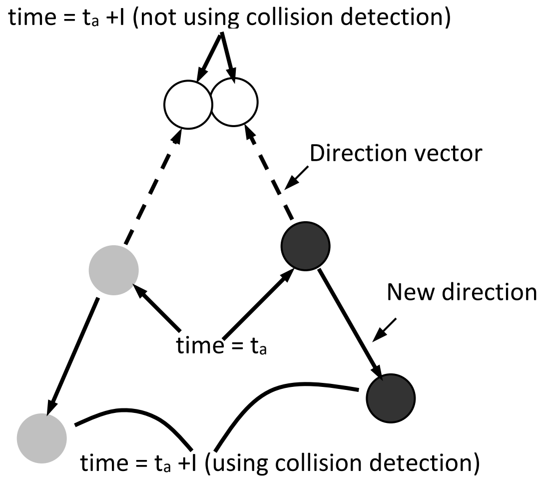


Figure 2: A particle-collision. The solid circles are the particle positions at time t_0 and t_{0+1} using collision detection. The dotted rings are their positions at time t_{0+1} without collision detection. The bounce-factor is set to -1.

The proposed Dimensioned Particle Swarm Optimization (DPSO) algorithm for solving reactive power optimization problem is given below.

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:

$$f_n = P^n{}_L + \alpha \sum_{j=1}^{NG} Q_{G,j}^{\lim,n} + \beta \sum_{j=1}^{NL} V_{L,j}^{\lim,n} \quad ; n = 1,2..,N_n \qquad \dots \qquad (9)$$

α, β = penalty factors

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

$$Q_{G,j}^{\lim,n} = \begin{cases} Q_{G,\min} - Q_{G,j}^n & \text{if } Q_{G,j}^n < Q_{G,\min} \\ Q_{G,j}^n - Q_{G,\max} & \text{if } Q_{G,j}^n > Q_{G,\max} \end{cases} \qquad \dots \qquad (10)$$

And

$$V_{L,j}^{\lim,n} = \begin{cases} \left|V_{L,j}^n\right| - V_{L,\max}, & \text{if } \left|V_{L,j}^n\right| > V_{L,\max} \\ 0 & \text{otherwise} \end{cases} \qquad \dots \qquad (11)$$

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 (7). The continuous equations are utilized for continuous variables and the discrete equations for discrete variables.

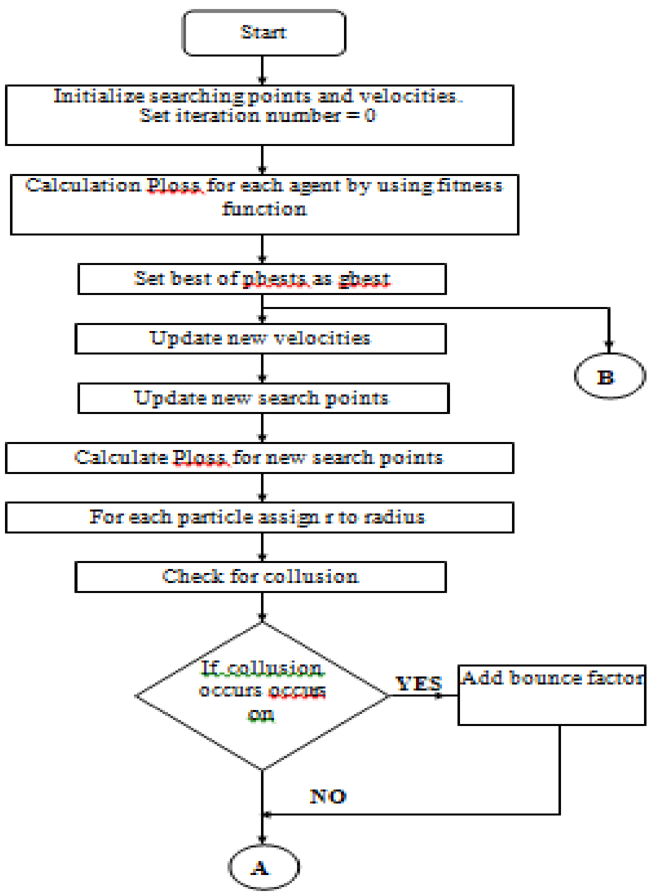
Step 5 New searching points are calculated using (8)**.** The continuous equations are utilized for continuous variables and the discrete equations for discrete variables.

Step 6 *P*loss 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 boundaries is largely weighted and added to the objective function as a penalty term.



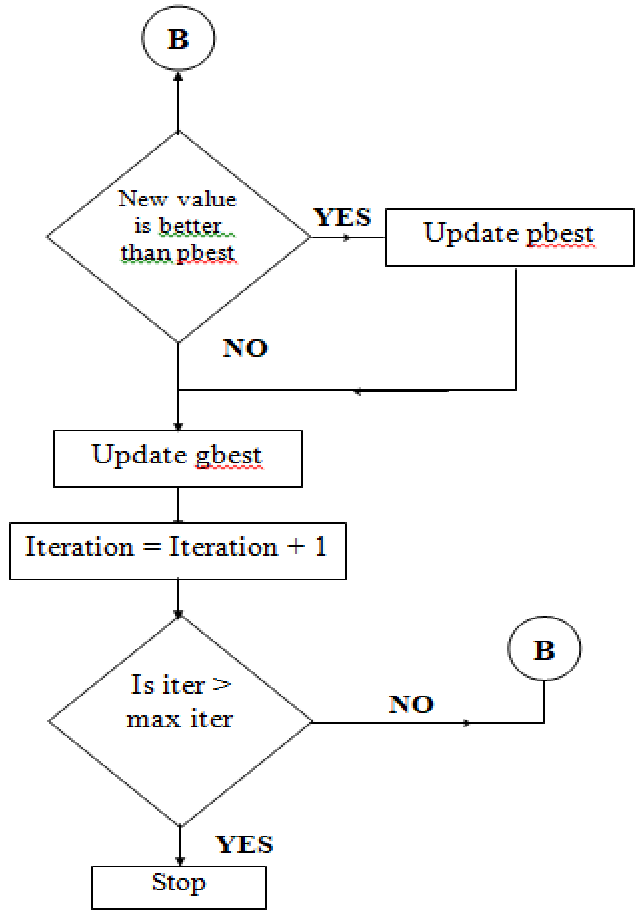


Figure 3: Flow chart of DPSO algorithm for RPO problem

5. Simulation Results

Dimensioned Particle Swarm Optimization (DPSO) algorithm has been tested in standard IEEE-57 bus power system. The reactive power compensation buses are 18, 25 and 53. Bus 2, 3, 6, 8, 9 and 12 are PV buses and bus 1 is selected as slack-bus. The system variable limits are given in Table 1.

The preliminary conditions for the IEEE-57 bus power system are given as follows:

P_{load} = 12.008 p.u. Q_{load} = 3.016 p.u.

The total initial generations and power losses are obtained as follows:

$\sum P_G$ = 12.112 p.u. $\sum Q_G$ = 3.3090 p.u.

P_{loss} = 0.25436 p.u. Q_{loss} = -1.2010 p.u.

Table 2 shows the various system control variables i.e. generator bus voltages, shunt capacitances and transformer tap settings obtained after optimization which are within the acceptable limits. In Table 3, shows the comparison of optimum results obtained from proposed methods with other optimization techniques. These results indicate the robustness of proposed approaches for providing better optimal solution in case of IEEE-57 bus system.

Table 1: Variable Limits

Reactive Power Generation Limits							
Bus no	1	2	3	6	8	9	12
Qgmin	-1.4	-.015	-.02	-0.04	-1.3	-0.03	-0.4
Qgmax	1	0.3	0.4	0.21	1	0.04	1.50

Voltage And Tap Setting Limits					
vgmin	**Vgmax**	**vpqmin**	**Vpqmax**	**tkmin**	**tkmax**
0.9	1.0	0.91	1.05	0.9	1.0

Shunt Capacitor Limits			
Bus no	18	25	53
Qcmin	0	0	0
Qcmax	10	5.2	6.1

Table 2: Control variables obtained after optimization

Control Variables	**DPSO**
V1	1.1
V2	1.022
V3	1.020
V6	1.018
V8	1.012
V9	1.000
V12	1.010
Qc18	0.0621
Qc25	0.100
Qc53	0.0210
T4-18	1.000
T21-20	1.000
T24-25	0.804
T24-26	0.802
T7-29	1.004
T34-32	0.800
T11-41	1.010
T15-45	1.030
T14-46	0.910
T10-51	1.010
T13-49	1.010
T11-43	0.910
T40-56	0.900
T39-57	0.950
T9-55	0.950

Table 3: Comparison results

S.No.	Optimization Algorithm	Finest Solution	Poorest Solution	Normal Solution
1	NLP [14]	0.25902	0.30854	0.27858
2	CGA [14]	0.25244	0.27507	0.26293
3	AGA [14]	0.24564	0.26671	0.25127
4	PSO-w [14]	0.24270	0.26152	0.24725
5	PSO-cf [14]	0.24280	0.26032	0.24698
6	CLPSO [14]	0.24515	0.24780	0.24673
7	SPSO-07 [14]	0.24430	0.25457	0.24752
8	L-DE [14]	0.27812	0.41909	0.33177
9	L-SACP-DE [14]	0.27915	0.36978	0.31032
10	L-SaDE [14]	0.24267	0.24391	0.24311
11	SOA [14]	0.24265	0.24280	0.24270
12	LM [15]	0.2484	0.2922	0.2641
13	MBEP1 [15]	0.2474	0.2848	0.2643
14	MBEP2 [15]	0.2482	0.283	0.2592
15	BES100 [15]	0.2438	0.263	0.2541
16	BES200 [15]	0.3417	0.2486	0.2443
17	Proposed DPSO	0.22054	0.23038	0.22266

6. Conclusion

In this paper Dimensioned Particle Swarm Optimization (DPSO) algorithm successfully solved Reactive power optimization (RPO) problem. Dimensioned extension is introduced to particles in the particle swarm optimization (PSO) model in order to overcome premature convergence in interactive optimization. In the performance of basic PSO often flattens out with a loss of diversity in the search space as resulting in local optimal solution. Proposed algorithm has been tested in standard IEEE 57 test bus system and compared to other standard algorithms. Simulation results reveal about the best performance of the proposed algorithm in reducing the real power loss and voltage profiles are within the limits.

References

[1] O.Alsac,and B. Scott, "Optimal load flow with steady state security",IEEE Transaction. PAS -1973, pp. 745-751.
[2] Lee K Y ,Paru Y M , Oritz J L –A united approach to optimal real and reactive power dispatch , IEEE Transactions on power Apparatus and systems 1985: PAS-104 : 1147-1153
[3] A.Monticelli , M .V.F Pereira ,and S. Granville , "Security constrained optimal power flow with post contingency corrective rescheduling" , IEEE Transactions on Power Systems :PWRS-2, No. 1, pp.175-182.,1987.
[4] Deeb N ,Shahidehpur S.M ,Linear reactive power optimization in a large power network using the decomposition approach. IEEE Transactions on power system 1990: 5(2) : 428-435
[5] E. Hobson ,'Network consrained reactive power control using linear programming, ' IEEE Transactions on power systems PAS -99 (4) ,pp 868=877, 1980
[6] K.Y Lee ,Y.M Park , and J.L Oritz, "Fuel –cost optimization for both real and reactive power dispatches" , IEE Proc; 131C,(3), pp.85-93.

(Received: Mar 09, 2018 - Accepted: Apr 26, 2018) DOI: 10.5281/zenodo.1248180

[7] M.K. Mangoli, and K.Y. Lee, “Optimal real and reactive power control using linear programming” , Electr.Power Syst.Res, Vol.26, pp.1-10,1993.

[8] K.Anburaja, “Optimal power flow using refined genetic algorithm”, Electr.Power Compon.Syst , Vol. 30, 1055-1063,2002.

[9] D. Devaraj, and B. Yeganarayana, “Genetic algorithm based optimal power flow for security enhancement”, IEE proc-Generation.Transmission and. Distribution; 152, 6 November 2005.

[10] Y.Tan and Z.M.Xiao, “Clonal particle swarm optimization and itsapplications,” Proc. Proceedings of the IEEE Congress on EvolutionaryComputation, Singapore, 2007, pp. 2303 - 2309.

[11] Yu Liu,Zheng Qin and Zhewen Shi, “Hybrid particle swarm optimizer with line search,” Proc. Proceedings of the IEEE International Conferenceon Systems, Man and Cybernetics, 2004, pp. 3751-3755.

[12] Junying Chen, Zheng Qin, Yu Liu and Jiang Lu, “Particle swarm optimization with local search,” Proc. Proceedings of the IEEE InternationalConference on Neural Networks and Brains, 2005, pp. 481-484.

[13] Mathew Mithra Noel and Thomas C.Jannett, “Simulation of a new hybrid particle swarm optimization algorithm,” Proc. Proceedings ofthe thirty-sixth Southeastern Symposium, 2004, pp. 150-153.

[14] Chaohua Dai, Weirong Chen, Yunfang Zhu, and Xuexia Zhang, “Seeker optimization algorithm for optimal reactive power dispatch,” IEEE Trans. Power Systems, Vol. 24, No. 3, August 2009, pp. 1218-1231.

[15] J. R. Gomes and 0. R. Saavedra, “Optimal reactive power dispatch using evolutionary computation: Extended algorithms,” IEE Proc.-Gener. Transm. Distrib.. Vol. 146, No. 6. Nov. 1999.

*Corresponding author.
E-mail address: gklenin@ gmail.com