Particle Swarm Optimization-based Algorithm for Optimal Reactive Power Dispatch

R. Menezes, P.M. Fonte, Rui Pestana
ISEL-Instituto Superior de Engenharia de Lisboa,
Instituto Politécnico de Lisboa
1950-005 Lisboa, Portugal
a42912@alunos.isel.pt, pedro.fonte@isel.pt, rui.pestana@isel.pt

2025 21st International Conference on the European Energy Market (EEM) | 979-8-3315-1278-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/EEM64765.2025.11050151

***Abstract* – Optimal Reactive Power Dispatch (ORPD) is very important for the security and economy of power systems. It is a mixed integer nonlinear optimization problem for which metaheuristic methods have proven to be effective in its solution. This paper presents an implementation of the particle swarm optimization algorithm (PSO) for the solution of the ORPD. Another approach, called Fitness-Distance Ratio PSO (FDR-PSO), is implemented to improve the results of the basic PSO. One other version called Second Order PSO (SO-PSO) is implemented for the same purpose. This second-order principle was combined with the FDR-PSO, making it the SO-FDR-PSO. These versions were tested on the IEEE 14 bus and IEEE 39 bus systems. The results show that the tested PSO algorithm and its improved versions can converge to good global optimal solutions (the best optimal solution the PSO could find), effectively solving the ORPD, with increasing degrees of success.**

***Index Terms*-- Nonlinear programming, Optimal reactive power dispatch, Particle swarm optimization**

I. Introduction

ORPD is a sub problem of optimal power flow calculation. Reactive power plays an important role in voltage stability and active power flow in a power system. The objectives of the ORPD are: minimization of active power losses; improvement of voltage profile; minimization of transmission costs and maximization of voltage stability in the system. This is achieved by proper adjustment of parameters of the power grid, mainly location and size of shunt capacitors/reactors, tap ratios of transformers and the reactive power output of generators, improving the quality of the grid's voltage profile and reducing power losses.

Several gradient-based optimization methods have been used to solve this problem, such as interior point methods, nonlinear programming, linear programming and quadratic programming. However, these methods solve nonlinear optimization problems only on an approximate basis [1]. Metaheuristic optimization methods, as a second-generation of solution techniques for the ORPD, have been proposed with the aim of overcoming the limitations of the gradient-based methods. Among these methods are genetic algorithms [2], particle swarm optimization [3], ant colony optimization [4], simulated annealing [5], fuzzy logic [6], among others. Meta-heuristic methods allow the search space to be explored more efficiently, having a greater probability of avoiding convergence to local minimum. However, such methods cannot guarantee that the obtained solution is the global optimum, but a solution very close to this point can often be found. The use of appropriate discretization methods and constraint handling techniques can guarantee that the (sub) optimal solution is always feasible [7]. This work intends to present a systematic analysis of different metaheuristics based on PSO and the way to apply these techniques to a problem with continuous and discrete variables. Departing from the basic PSO, followed by the SO-PSO, FDR-PSO and finishing combining the previous methods resulting in the SO-FDR-PSO.

This paper is organized as follows: in section II the mathematical model of the ORPD is defined; in section III it is explained the concept of particle swarm optimization, and it is presented its mathematical model; in section IV describes the developed PSO algorithm; in section V the case studies for developed algorithms are presented, and an analysis of the results is conducted. In section VI the most relevant conclusions are summarized, and perspectives of futures work are discussed.

II. Matemathical Modelling of the ORPD

A. Objective function and restrictions

The ORPD is generally formulated as single-objective optimization problem to minimize active power losses (transmission losses) in a power system. The control variables considered are the generator (PV bus) voltages (V_G), the ratio of transformer taps (T_k) and shunt capacitor reactive power outputs (Q_c).

The objective function for active power losses in a power system is formulated as (1):

$$\min P_{loss} = \sum_{k=1}^{N_{BR}} G_k \left[V_i^2 - V_j^2 - 2V_iV_j \cos\left(\delta_i - \delta_j\right)\right] \tag{1}$$

Where k=1,2,…,N_{Br}, being N_{Br} the total number of branches, P_{loss} is the total active power loss in the transmission lines, G_k is the conductance of branch k between buses i and j, V_i , V_j are the voltage magnitudes at buses i and j and δ_i , δ_j are the load angles at buses i and j.

The ORPD must comply with the power balance of the system as well as its operating limits. The power balance equations ensure that the load demand is met considering the power losses in the system and are accounted during the power flow solution. These equations constitute the equality constrains of the problem and are calculated by (2) and (3)

$$P_{G_i} - P_{D_i} - V_i \sum_{j=1}^{N} V_j \left[G_{ij} \cos\left(\delta_i - \delta_j\right) + B_{ij} sen\left(\delta_i - \delta_j\right) \right] = 0 \quad (2)$$

$$Q_{G_i} - Q_{D_i} - V_i \sum_{j=1}^{N} V_j \left[G_{ij} sen\left(\delta_i - \delta_j\right) - B_{ij} \cos\left(\delta_i - \delta_j\right) \right] = 0 \quad (3)$$

where $j = 1,2,\ldots, N$, being N the total number of buses; P_{Gi} and Q_{Gi} are the active and reactive power of the generators connected to the i-th bus, respectively; P_{Di} and Q_{Di} are the active and reactive power of the load connected to the i-th bus, and G_{ij} and B_{ij} are the branch conductance and susceptance between bus i and bus j, respectively.

The limits of the control variables, generator (PV bus) voltages (V_G), the ratios of transformer taps (T_k) and shunt capacitor reactive power outputs (Q_C), constitute inequality constraints defined by (4),(5) and (6).

$$V_{G_i}^{\min} \le V_{G_i} \le V_{G_i}^{\max} \quad (4)$$

$$T_k^{\min} \le T_k \le T_k^{\max} \quad (5)$$

$$Q_{C_i}^{\min} \le Q_{C_i} \le Q_{C_i}^{\max} \quad (6)$$

The limits of system state variables, which consist of load bus voltages (V_i), reactive power of generators (Q_{Gi}), and line power flow (S_{lm}) are also inequality constraints defined by (7), (8) and (9).

$$V_i^{\min} \le V_i \le V_i^{\max} \quad (7)$$

$$Q_{G_i}^{\min} \le Q_{G_i} \le Q_{G_i}^{\max} \quad (8)$$

$$S_{lm}^{\min} \le S_{lm} \le S_{lm}^{\max} \quad (9)$$

The satisfaction of these constraints is essential for a stable and secure functioning of the power system.

III. Particle Swarm Optimization

In the continuous space coordinate system, the PSO is mathematically described in the following manner: assuming a swarm size of N, each particle's position vector in D-dimensional space is $X_i = (x_i^1, x_i^2, \ldots, x_i^d)$, and its velocity vector is $V_i = (v_i^1, v_i^2, \ldots, v_i^d)$. A particle's individual best position is $P_i = (p_i^1, p_i^2, \ldots, p_i^d)$, and the global best position is $P_g = (p_g^1, p_g^2, \ldots, p_g^d)$. In the original version of the algorithm, for a minimization problem, the individual best positions are updated as (10).

$$p_{i,t+1}^d = \begin{cases} x_{i,t+1}^d & , f\left(X_{i,t+1}\right) < f\left(P_{i,t}\right) \\ p_{i,t}^d & , otherwise \end{cases} \quad (10)$$

The velocity and position of each particle is updated by (11) and (12).

$$v_{i,t+1}^d = \omega . v_{i,t}^d + c_1 . rand_1 \left(p_{i,t}^d - x_{i,t}^d\right) + c_2 . rand_2 \left(p_{g,t}^d - x_{i,t}^d\right) \quad (11)$$

$$x_{i,t+1}^d = x_{i,t}^d + v_{i,t+1}^d \quad (12)$$

Where ω, designated inertia weight, is the parameter that affects the degree of influence of the previous velocity, *rand* is a random value in the range [0, 1] and c_1 and c_2 are stochastic acceleration terms that attract each particle towards its individual best, *pBest* (p_i) and the global best, *gBest* (p_g), respectively. It's important to select appropriate values for these parameters as the performance of the PSO is highly affected by them.

The PSO displays several advantages, which are, summarily: The objective function is not required to be differentiable nor continuous; its convergence speed and rate are high and is easy to implement through programming. Like all the other metaheuristic algorithms, the PSO algorithm has disadvantages: When applied to objective functions with multiple local optima, it has a chance to converge to a local optimum. The main reason for it happening is the disappearance of diversity within the particle population, which causes premature convergence of the algorithm to a local optimum [8]. Therefore, it is important to take measures to increase diversity of the particles.

Metaheuristic algorithms such as the PSO realize an unconstrained search according to their definitions. Therefore, when applied to constrained optimization problems, these algorithms require additional constraint handling methods. By using a constraint handling method, the information of infeasible individuals can be exploited, resulting in better solutions.

The simplest and early constraint method in literature is the penalty function approach [9]. The effectiveness of the penalty function depends on the adequate selection of the penalty coefficients, which are generally found by trial and error, which is tedious and prone to error. Various constraint handling methods that require no manual parameter adjustment have been proposed in literature, such as the principle of constrained non-domination.

According to the principle of constrained non-domination defined in [10], candidate solution X_i is said to dominate candidate solution X if and only if one of the following scenarios occur:

- X_i is feasible and X_j is infeasible;
- Both solutions are infeasible, and X_i has a smaller overall constraint violation than X_j;
- Both solutions are feasible, and X_i is not worse than X_j in objective function value.

IV. Solution Methodolgy

A. Problem space coding

The position of a particle i, or a possible solution, is a multi-dimensional vector composed by the control variables: reactive power output of shunt capacitor banks, transformer tap ratios, and generator voltages as in (13),

$$X_i = \left[\left(Q_{C_1}, Q_{C_2}, \ldots, Q_{C_{d_c}}\right), \left(T_{k_1}, T_{k_2}, \ldots, T_{k_{d_t}}\right), \left(V_{G_1}, V_{G_2}, \ldots, V_{G_{dg}}\right) \right] \quad (13)$$

where d_c is the number of shunt capacitors banks in the system, d_t is the number of transformers with tap regulation, and d_g is the number of generators in the system.

The shunt capacitor reactive power Q_c and transformer tap positions T_k are discrete control variables which can assume one of the values allowed by the step n of the capacitor banks and transformer tap positions respectively, forming the vector of discrete values X_{dis_i}, defined by (14),

$$X_{dis_i} = \left[x_{\min}, x_{\min} + n, x_{\min} + 2n, ..., x_{\min} + (m-1)n \right] \quad (14)$$

where x_{min} and x_{max} are the lower limit of the discrete variable X_i and $m = \frac{x_{max} - x_{min}}{n}$.

The state variable vector is composed of the generators reactive power, load bus voltages and line power flow, as (15),

$$Y_j = \left[\left(Q_{G_1}, Q_{G_2}, ..., Q_{G_{d_g}} \right), \left(V_1, V_2, ..., V_{d_b} \right), \left(S_{lm_1}, S_{lm_2}, ..., S_{lm_{d_l}} \right) \right] \quad (15)$$

where d_g is the number of generators, d_b is the number of buses and d_l is the number of transmission lines.

B. Parameters

The inertia weight was set to decrease linearly along the iterations with expression (16),

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{iter_{\max}} iter \quad (16)$$

where $\omega_{max} = 0.9$ and $\omega_{min} = 0.4$, $iter_{max}$ is the maximum number of iterations and $iter$ is the number of the current iteration. This means the inertia weight decreases linearly from 0.9 to 0.4. These values allow a balance between global and local exploration. The acceleration factors c_1 and c_2 were both set to 1, meaning the same weight is given to cognitive and social search. The constriction factor χ was proposed by [11], which usually improves the convergence rate, results from (17).

$$\chi = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|} \quad (17)$$
$$\varphi = c_1 + c_2$$

The velocity update defined by (11) becomes as (18).

$$v_{i,t+1}^d = \chi \cdot \left[\omega . v_{i,t}^d + c_1 . rand_1 \left(p_{i,t}^d - x_{i,t}^d \right) + c_2 . rand_2 \left(p_{g,t}^d - x_{i,t}^d \right) \right] \quad (18)$$

The number of particles was adjusted for each power system in study by experimentation, to achieve a balance between convergence speed and quality of solutions. The position and velocity of each particle were initialized randomly in each dimension, within the bounds of the respective control variables.

C. Constraint handling

The equality (2) and (3) are met by the power flow solution method, in this case the Newton-Raphson method. The inequality constraints for the control variables (4), (5) and (6) (V_G, T_k, Q_c) are met by imposing maximum and minimum position limits, after, the position update equation is applied to each particle with position X_i.

The inequality constraints for the state variables (7), (8) and (9), (V_i, Q_G, S_{lm}), as they are dependent variables, are handled with the following method: after the power flow is solved and the objective function value (active power loss) is obtained for each particle, the violation of each inequality constraint G_i is calculated for each state variable of vector Y_j. Then, the overall constraint violation $\upsilon(Y_j)$ is calculated for each state variable vector according with (19), where w_i is a weight parameter given by $w_i = \frac{1}{G_{max_i}}$ and $G_{\max_i}$ is the maximum constraint violation for each constraint.

$$\upsilon(Y_i) = \frac{\sum_{i=1}^{m} w_i G_i(Y_i)}{\sum_{i=1}^{m} w_i} \quad (19)$$

$$G_{\max_i}(X) = \max\{g_i(X), 0\} \quad , i = 1, 2, ..., p$$

Next, the principle of constrained non-domination is applied to the individual best position P_i update with (20),

$$p_{i,t+1}^d = \begin{cases} x_{i,t+1}^d & , f(X_{i,t+1}) < f(P_{i,t}) \text{ and } \upsilon(Y_{i,t+1}) \le \upsilon(P_{Yi,t}) \\ x_{i,t+1}^d & , \upsilon(Y_{i,t+1}) \le \upsilon(P_{Yi,t}) \\ p_{i,t}^d & \text{otherwise} \end{cases} \quad (20)$$

where P_{Y_i} is the state variable vector obtained with the individual best position P_i. Expression (20) means that the current position will always become the new individual best if it has a smaller overall constraint violation, regardless of the objective function value. In other words, a feasible solution is always chosen over an infeasible solution. This results in guaranteed convergence to a feasible solution, given enough iterations.

D. Updating discrete variables

The PSO performs a search on the continuous domain. After updating the position of each particle, the final step of the algorithm is to approximate the control variables Q_c and T_k to their nearest feasible discrete domain locations. The two discrete vectors which constitute the discrete domains are:

$$Q_{C_{dis}} = [Q_1, Q_2, ..., Q_m]$$
$$T_{k_{dis}} = [T_1, T_2, ..., T_m]$$

The method implemented to update the discrete variables was proposed by [7]. The location of a particle in the discrete domain is defined by a local hypercube expressed as (21),

$$H_d = \left[\left(x_1^L, x_1^U \right), \left(x_2^L, x_2^U \right), \ldots, \left(x_m^L, x_m^U \right) \right] \tag{21}$$

$$x_i^L < x_i < x_i^U, \qquad i = 1, 2, \ldots, m \tag{22}$$

where m is the number of discrete variables, x_i a variable in the current particle position, and x_i^L and x_i^U are two consecutive values in a discrete vector X, which define the vertices of the local hypercube. The total number of vertices in the hypercube is then 2^m.

The Nearest Vertex Approach (NVA) is used to approximate each variable x_i (Q_c and T_k) of the current position to the nearest vertex of its local hypercube H_d based on Euclidean distance, as in (23).

$$\tilde{x}_i = \begin{cases} x_i^L & , \left| x_i - x_i^L \right| \le \left| x_i - x_i^U \right| \\ x_i^U & , otherwise \end{cases}, i = 1, 2, \ldots, m \tag{23}$$

With this approximation, the control variables Q_c and T_k in the current solution will become discrete values allowed by the respective discrete vectors

$$\tilde{X} = \left| \tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_m \right|.$$

E. Fitness-Distance Ratio PSO

It has been seen that the global PSO has the tendency to get trapped in a local optimum, essentially due to all particles following the same global best position. Reference [12] addressed this issue by proposing a mixed global-local PSO, called FDR-PSO. In this algorithm, one particle for each velocity dimension is selected, from their neighborhood, to update the velocity of each particle to counter the possibility of the movement towards different particles cancelling each other. The neighborhood of each particle consists of a maximum of five particles with a saved individual best position. At each iteration, a new particle is appended to the neighborhood while the oldest particle is removed. The selected particles must satisfy two criteria: they must be near the current particle, and they must have visited a position of higher fitness. As such, a particle from the neighborhood with individual best position P_j is selected such that expression (24) is maximized.

$$\frac{f\left(P_j\right) - f\left(X_i\right)}{\left| P_j^d - X_j^d \right|} \tag{24}$$

The particle with individual best P_j that maximizes this ratio will become the neighborhood best (*nBest)* particle P_{n_i}, towards which the dimension d of the velocity of particle i is updated. The influence of the *gBest* is maintained for higher diversity of particles, therefore a new component is added in the velocity update equation for the *nBest.* The velocity update equation results from (25).

$$\begin{aligned} v_{i,t+1}^d = \chi.[\omega.v_{i,t}^d + c_1.rand_1\left(p_{i,t}^d - x_{i,t}^d\right) + \\ + c_2.rand_2\left(p_{g,t}^d - x_{i,t}^d\right) + c_3.rand_3\left(p_{n_i,t}^d - x_{i,t}^d\right)] \end{aligned} \tag{25}$$

The acceleration factors c_1, c_2 and c_3 were all set to 1. By making the acceleration factors equal each particle is equally pulled towards the *pBest*, *gBest* and *nBest* particles. The FDR-PSO was implemented as an optional function for the global PSO, hence the constraint handling and updating of discrete variables are also maintained.

F. Second Order PSO

One method to increase performance is by improving the initialization of the particles, as noted by [13], where is proposed a variation of the PSO that optimizes the initialization of the particles, called Second Order PSO (SO-PSO). The second order PSO can be described as running the PSO algorithm more than once, where the initial positions for the second run are generated based on the global best position obtained at the end of the first run, maintaining this global best position as the initial global best position. In this manner the algorithm continues the search for an optimum after convergence, having a good solution as a starting point.

The initial position for each particle is generated by making each control variable in the final *gBest* position vary randomly within their minimum and maximum change limit with (26) and (27),

$$\min \lim_{x_i^d} = \frac{x_{\min}^d}{p_{g\min}^d} \tag{26}$$

$$\max \lim_{x_i^d} = \frac{x_{\max}^d}{p_{g\max}^d} \tag{27}$$

where x_{min}^d and x_{max}^d are the lower and upper limit of the control variable at dimension d and $p_{g_{min}}^d$ and $p_{g_{max}}^d$ are the minimum and maximum value of the control variable at dimension *d* found at the *gBest* position respectively.

Finally, the new initial positions for the next run are calculated with (28), for each particle *i*.

$$\begin{aligned} x_{init} = p_g^d.rand_i^d \\ rand_i^d \in \left[\min\lim_{x_i^d}, \max\lim_{x_i^d} \right] \end{aligned} \tag{28}$$

The stopping criterion of the implemented SO-PSO is defined as a maximum number of runs of the global PSO. The maximum number of runs was set to 10. By experimentation, this stopping criterion was found sufficient to obtain a good optimal solution at convergence in the final run. The SO-PSO is also implemented as an optional function for the global PSO. In this manner the SO-PSO can be combined with the FDR-PSO, hence designated SO-FDR-PSO.

The power systems for the testing of the algorithm were implemented in Siemens PSS E Xplore 34. The algorithm was written in Python 2.7, and the code interacts with PSS E through its API, which allows the changing of the control variables in the system, solving the power flow, reading the state variables and calculating the power losses.

V. Case Studies

A. Test systems data and parametrs

The proposed versions of the PSO in the previous chapter were tested on the IEEE 14 bus and IEEE 39 bus systems.

The IEEE 14 bus system has five generators, at buses 1, 2, 3, 6 and 8, three transformers with tap regulation between buses 4-7, 4-9 and 5-6, and one shunt capacitor at bus 9. The voltage limits on all buses are [0.9, 1.1] p.u, the tap step of the regulating transformer is 0.0125 p.u and the tap position limits are [0.9, 1.1] p.u. The shunt capacitor battery has three equal groups, each one with 6 Mvar.

The IEEE 39 bus system has ten generators, from buses 30 up to 39, one transformer with tap regulation between buses 6-31, two shunt capacitors at buses 4 and 5. The voltage limits on all buses are [0.9, 1.1] p.u;, the step of the transformer tap positions and the tap position limits are the same as in the IEEE 14 bus system; The shunt capacitor battery at bus 4 has four equal groups, each one with 25 Mvar, and at bus 5 has eight groups, each one with 25 Mvar.

For the IEEE 14 bus system, a maximum of 50 iterations of the PSO were sufficient for convergence. The number of particles was set to 60. The SO-PSO has a maximum of 10 iterations, adding up 500 iterations for the PSO.

For the IEEE 39 bus system, a maximum of 100 iterations of the algorithm were sufficient for convergence. The number of particles was set to 90. The SO-PSO has a maximum of 10 iterations, adding up 1000 iterations for the PSO.

B. Results and analysis

A statistics-based analysis is required to evaluate the performance of a stochastic algorithm such as the PSO. The PSO, FDR-PSO, SO-PSO and SO-FDR-PSO were run 30 times for the IEEE 14 bus system and IEEE 39 bus system. The performance of the algorithms was measured with the following indicators:

- Average accuracy: the average objective function value (active power losses, in MW) of the 30 runs;
- Standard deviation (consistency) in relation to the average accuracy of the 30 runs;
- Best accuracy: the best objective function value (lowest active power losses) obtained within 30 runs of the algorithm;
- Success rate: percentage of the total runs where the best accuracy was reached. Two decimal places in the objective function value were considered in calculating the success rate;
- Convergence time: the approximate average time elapsed in each run.

The performance indicators obtained with the results of 30 runs of the algorithm are displayed in Tables I and II.

Foremost, it was verified that all the control variables were within their respective bounds, also the discrete control variables, Q_c and T_k assumed one of the values from their discrete vectors. Therefore, all the obtained solutions are feasible both in the continuous domain and the discrete domain. All the state variables obtained with each solution are also in their feasible domain, which means that the constraint handling methods employed, and the nearest vertex approach are effective in this problem. Tables III and IV display an example of an optimal solution to IEEE14 and IEEE39.

TABLE I. Performance indicators for IEEE 14 bus

Performance indicators				
	PSO	***FDR-PSO***	***SO-PSO***	***SO-FDR-PSO***
Avg. accuracy [MW]	12.387	12.297	12.298	12.280
Std. dev.	0.066	0.015	0.030	0.000
Best accuracy [MW]	12.280	12.281	12.280	12.280
Success [%]	7	17	57	100
Convergence Time [s]	28	140	280	1400

TABLE II. Performance indicators for IEEE 39 bus

Performance indicators				
	PSO	***FDR-PSO***	***SO-PSO***	***SO-FDR-PSO***
Avg. accuracy [MW]	38.134	38.116	38.006	37.955
Std. dev.	0.197	0.097	0.036	0.007
Best accuracy [MW]	37.962	37.975	37.948	37.949
Success [%]	3	3	7	63
Convergence Time [s]	93	700	930	7000

TABLE III. Example of optimal solution in the 1st run of the global PSO on the IEEE 14 bus

Q_C	18
T_K	[1.0125, 0.9, 0.9875]
V_G	[1.1, 1.086, 1.1, 1.1, 1.1]

TABLE IV. Example of optimal solution in the 1st run of the global PSO on the IEEE 39 bus

Q_C	25, 50
T_K	[0.95]
V_G	[1.041, 1.1, 1.037, 1.042, 1.051,1.094, 1.1, 1.072, 1.07,1.078]

For the 14 buses system, through comparison of the performance indicators, it can verify that the FDR-PSO has better consistency and average accuracy than the global PSO. And knowing that in theory the FDR-PSO decreases the probability of premature convergence to local optima thereby improving the optimal solutions, we can conclude the FDR-PSO behaved as expected. We can infer that better average accuracy, success rate, and consistency translate to lower probability of premature convergence to local optima. As such, for the SO-PSO, given the slightly improved average accuracy and much superior success rate and consistency, there appears to be a very low chance of premature convergence to a local optimum. The SO-FDR-PSO further enhances these results, to the extent that there is no standard deviation and a success of 100%, meaning the SO-FDR-PSO consistently reached the best solution obtained so far. Through considerate observation of the results of one run of the SO-PSO in Table V we can assume that the entire search space was explored, to the best of the capability of the PSO.As indicated in section IV-F, if a better gBest is found with the SO-PSO initialization mechanism, the optimization process will continue as the particles will now have a new position to follow. We can see that through this mechanism, the swarm cannot find an optimal solution that results in losses lower than exactly 12.27964 MW, leading us to conclude that the searching capability of this PSO was exhausted, and that this optimal solution is possibly the global optimum of the problem, or at least very close to it.

TABLE V. DETAILED RESULTS OF ONE OF SO-PSO AND SO-FDS-PSO FOR THE IEEE 14 BUS

Optimized power losses [MW]		
Run nº	***SO-PSO***	***SO-FDR-PSO***
1	12,457	12,281
2	12,382	12,279
3	12,339	12,279
4	12,339	12,279
5	12,339	12,279
6	12,280	12,279
7	12,279	12,279
8	12,279	12,279
9	12,279	12,279
10	12,279	12,279

In the case of the 39 buses system, the FDR-PSO remains clearly more consistent than the global PSO, although the other performance indicators are identical, meaning that we cannot affirm that the FDR-PSO will certainly not converge prematurely to local optima, but we can affirm that it has a smaller probability to do so. The overall advantage of the SO-FDR-PSO stands as observed by the clearly superior performance indicators.By making the same observations done for the 14 buses system, regarding the search space and the best optimal solution, for the 39 buses system the global optimum should be close to approximately 37.948 MW, as no better solution could be reached with the SO-PSO.

VI. CONCLUSION

In this paper a PSO algorithm was developed for the minimization of active power losses within the ORPD and was tested on the IEEE 14 bus and IEEE 39 bus systems. The algorithm proved to be effective as it always converged to solutions, which were always feasible both in the continuous domain and the discrete domain. A comparison of the results obtained with these versions of the PSO was made through a statistics-based analysis, and it was found that the FDR-PSO obtained improved solutions as expected. The FDR-PSO consistently avoided premature convergence to local optima, solving the primary issue of the PSO, allowing us to conclude that the FDR-PSO is superior to the PSO when applied to the ORPD. The SO-PSO, and particularly the SO-FDR-PSO presented the most noteworthy results. In a practical setting, the convergence time may have to be considered, and certain parameters will have to be adjusted such as neighbourhood size, number of iterations of the SO-PSO or a different stopping criterion, so that a compromise is reached between convergence time and quality of the optimal solutions.

This version of the algorithm consistently converged to what appears to be the global optimum of the problem. Future work will focus on solving the ORPD with other objective functions such as voltage profile improvement, voltage stability improvement or system costs minimization. Attention will be given particularly to the SO-PSO since it has proven to be promising.

REFERENCES

[1] Fan, Zheng, Chang, Xiqiang, Wang, Heng, Pu, Tianjiao, Yu, Ting, Liu, Guangyi, "Discrete Reactive Power Optimization Based on Interior Point Filter Algorithm and Complementarity Theory", *POWERCON 2014 - 2014 International Conference on Power System Technology: Towards Green, Efficient and Smart Power System*, pp. 210-214, 2014

[2] L. Rojas, R. Garcia, L. Roa, "Optimal Capacitor Location for Radial Systems using Genetic Algorithms", in *Proc. 2006 IEEE PES Transmission and Distribution Conference and Exposition: Latin America, TDC'06*, pp. 1-4, 2006;

[3] Z. Wang, L. Li and B. Li, "Based on Particle Swarm Optimization and Simulated Annealing Combined Algorithm for Reactive Power Optimization," *2009 Asia-Pacific Power and Energy Engineering Conference*, Wuhan, China, 2009, pp. 1-5

[4] Kasaei, M. J.; Gandomkar, M., "Loss Reduction in Distribution Network Using Simultaneous Capacitor Placement and Reconfiguration With Ant Colony Algorithm", in *Proc. Asia-Pacific Power and Energy Engineering Conference, APPEEC*, pp. 10-13, 2010;

[5] Guo, Liya; Ding, Xiaoqun; Chen, Guangyu, "A Combination Strategy for Reactive Power Optimization Based on Model of Soft Constrain Considered Interior Point Method and Genetic-Simulated Annealing Algorithm", in *Proc. 2010 International Conference of Information Science and Management Engineering, ISME 2010*, pp. 151-154, 2010

[6] Bhattaeharya, S. K.; Goswami, S. K. , "Improved fuzzy based capacitor placement method for radial distribution system", in *Proc. 2008 Joint International Conference on Power System Technology POWERCON and IEEE Power India Conference, POWERCON*, pp. 1-5, 2008

[7] Chowdhury, S., Tong, W., Messac, A., Zhang, J., "A mixed-discrete Particle Swarm Optimization algorithm with explicit diversity preservation", in *Proc. Structural and Multidisciplinary Optimization*, 2013, pp. 367-388,

[8] Kennedy J, Eberhart RC., "Particle swarm optimization?", in *Proc. IEEE international conference on neural networks*, pp. 1942–1948, Perth, Australia

[9] K. Deb, "An efficient constraint handling method for genetic algorithms", in *Proc. Computer Methods in Applied Mechanics and Engineering 186* (2000), pp. 311–338

[10] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in *Proc. IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002

[11] Shi Y, Eberhart RC (1998), "A modified particle swarm optimizer.", in *Proc.* IEEE international conference on evolutionary computation, pp 69–73, Anchorage, Alaska, USA,May 4–9, 1998

[12] Peram T, Veeramachaneni k, Mohan CK, "Fitness-distance-ratio based particle swarm optimization." in *Proc. 2003 IEEE swarm intelligence symposium*, pp 174–181, Indianapolis, Indiana, USA, April 24–26, 2003

[13] M. Y. B. Prasanna, T. R. N. Balaji and S. M. Kannan, "Maximization of annual savings by using fuzzy-Second order PSO based optimal capacitor placement in RDF for constant load," *2012 IEEE Students' Conference on Electrical, Electronics and Computer Science*, Bhopal, India, 2012, pp. 1-4