ENHANCED PARTICLE SWARM OPTIMIZATION METHOD FOR POWER LOSS REDUCTION IN DISTRIBUTION SYSTEMS

Mihai GAVRILAS
Technical University, Iasi – Romania
mgavril@ee.tuiasi.ro

Ovidiu IVANOV
Technical University, Iasi – Romania
ovidiuivanov@ee.tuiasi.ro

Calin Viorel SFINTES
Eximprod S.A. - Romania
calin.sfintes@eximprod.ro

ABSTRACT

This paper presents a new approach to the reactive power compensation in distribution systems using an enhanced version of the Particle Swarm Optimization algorithm. Second order correction terms are used to compute the velocity of each particle during the search process. The new approach has better convergence properties.

INTRODUCTION

One of the major concerns of a power distribution company (DISCO) is to reduce technical and non-technical losses, while ensuring that the demand is satisfied at any moment at a reliable level of system performance and with the lowest possible cost. At present, due to the continuous growth of load demands, the distribution systems operate closer to their limits. At the same time the liberalization of electricity markets drives them to decision making based more and more on electricity prices.

Both aspects are influenced in a great extant by the value of technical losses in distribution networks. Some of the most efficient methods applied to reduce losses in distribution systems are: (a) balancing loads between the phases of the 3-phase system in the low voltage networks; (b) reconfiguring the distribution network in normal and post-outage conditions by changing its radial structure, and (c) controlling the reactive power to obtain desired voltage profiles and to change reactive power flow through the system.

Reactive power control at the system level can be achieved using several approaches such as generator voltage control, transformer tap control and fixed or controllable VAR sources. At the distribution level, the most efficient approach, as it produces other positive effects too, is the Reactive Power Compensation (RPC) method through power factor correction.

Current and power flows in distribution systems reduce through RPC. For instance, a reasonable 10% reduction in the current flow produces a 19% reduction in branch losses. Therefore, the RPC approach might be an important source for loss reduction and financial savings for any DISCO. In addition the RPC approach based on capacitor banks determines reduced voltage drops in the system and decreases voltage fluctuations.

This paper considers the RPC problem as finding an appropriate placement of reactive power sources (capacitor banks) to minimize system losses, while ensuring predefined voltage profiles in the buses of the distribution system. Inequality constraints are also considered to account for minimum and maximum voltage levels or the maximum allowable power supplied by reactive sources.

Traditional techniques use linear or/and non-linear programming or gradient descent optimization methods. The last two decades have shown an impressive development of new methods and algorithms based on stochastic methods and computational intelligence. These approaches have widely used simulated annealing, genetic programming or genetic algorithms [3,6]. Recently new computational intelligence approaches, based on immune algorithms [5] and particle swarm optimization techniques [2], were proposed. All these techniques aim to solve complex optimization problems by applying models and mechanisms inspired from natural selection, immune systems or swarm behavior.

This paper presents a comparative study for the RPC problem using three optimization methods based on computational intelligence techniques, namely genetic algorithms (AGs), immune algorithms (IAs) and particle swarm optimization (PSO) algorithms. A new searching strategy is also proposed to enhance the convergence properties of the standard PSO (S-PSO) algorithm.

PROBLEM FORMULATION

The RPC problem through optimal placement of reactive sources aims to identify an optimal solution for the placement of a stock of capacitors in the nodes of a distribution system with a known configuration that minimize an objective function. The unknown variables are locations, type and ratings of capacitors. This problem is one of the most complex optimization problems in distribution systems as it acts on an essentially non-linear objective function (power losses or combination of power losses and other functions for multiple criteria optimization) and must comply with non-linear inequality constraints.

The objective function addresses three optimization criteria: (i) minimizing system energy losses during a given period of time; (ii) minimizing voltage deviations with respect to the system rated voltage and (iii) minimizing the number of shunt capacitors installed in the system.

On the other hand, three types of constraints must be controlled: (i) keeping the system voltage between minimum and maximum values; (ii) excess reactive power compensation in the nodes of the distribution system is forbidden and (iii) number of capacitors installed in the system is limited by an existing stock.

The optimization criteria and the constraints are included in a unique objective function:

$$F_{obj} = \alpha_W \cdot F_W + \alpha_U \cdot F_U + \beta_U \cdot P_U + \beta_C \cdot P_C \qquad (1)$$

where: F_{obj} – the global optimization function of the problem; F_W – partial objective function for the system energy losses; F_U – partial objective function for the absolute voltage deviations with respect to the rated voltage; P_U –penalty function for minimum and maximum voltage constraints; P_C – penalty function for compensation constraints; α_W, α_U, β_U, β_C – weighting coefficients. Partial objective functions and penalty functions are expressed in specific percentage values referenced to the number of nodes in the system N and the time, in hours H.

The voltage penalty function P_U has two components:

$$P_U = P_U^{\min} + P_U^{\max} \tag{2}$$

associated to the minimum and maximum voltage constraints. The compensation strategy penalty function P_C has also two components:

$$P_C = P_C^{Q\max} + P_C^{N\max} \tag{3}$$

where: P_C^{Qmax} is the penalty function for the excess compensation of reactive power, and P_C^{Nmax} is the penalty function that covers both minimization of number of shunt capacitors installed in the system and the upper limit constraint of this value.

The terms from eqs. (1), (2) and (3) are computed as:

$$F_W = \frac{\sum_h dW_h}{\sum_h \sum_i P_{i,h}} \cdot 100 \tag{4}$$

$$F_U = \sum_h \sum_i \frac{\left|U_{i,h} - U_{nom}\right|}{U_{nom}} \cdot 100 \cdot \frac{1}{N \cdot H} \tag{5}$$

$$P_U^{\min} = \sum_h \sum_i \frac{\left\langle U_{\min} - U_{i,h}, 0\right\rangle}{U_{nom}} \cdot 100 \cdot \frac{1}{N \cdot H} \tag{6}$$

$$P_U^{\max} = \sum_h \sum_i \frac{\left\langle U_{i,h} - U_{\max}, 0\right\rangle}{U_{nom}} \cdot 100 \cdot \frac{1}{N \cdot H} \tag{7}$$

$$P_C^{Q\max} = \sum_h \left[\frac{\sum_i \left\langle QK_i - Q_{i,h}, 0\right\rangle}{\sum_i \left\langle Q_{i,h}, 0\right\rangle} \right] \cdot 100 \tag{8}$$

$$P_C^{N\max} = \frac{\sum_i NK_i - NCB}{NCB} \cdot 100 \tag{9}$$

where: $P_{i,h}$, $Q_{i,h}$ – active and reactive power load in node i at hour h; dW_h – system energy losses at hour h; $U_{i,h}$ – voltage magnitude in node i at hour h; U_{nom} – system rated voltage; U_{nin} , U_{nax} – minimum and maximum admissible voltages; QK_i – reactive power of shunt capacitors from node i; NK_i – number of shunt capacitors from nod i; NCB – total number of available capacitor banks. Also, the notation $<X,0>$ stands for X if X is positive and 0 otherwise.

Active and reactive powers in the system nodes are modelled using typical load profiles (TLPs) associated to known consumer categories from the system [4]. For instance, for a generic node i whose category is described by an active TLP denoted $P^*_{I,h}$ [%] (h=1,…,24), expressed in percentage from the daily energy consumption, the real load profile was computed using as reference the energy consumption over a given time - ND days – denoted W_{tot}:

$$P_{i,h} = \frac{P^*{}_{i,h}[\%]}{100} \cdot \frac{W_{tot}}{ND} \tag{10}$$

OPTIMIZATION MODELS

This paper approaches the RPC problem using three optimization models based on computational intelligence techniques. All these techniques have in common the fact that they can be applied to problems whose solutions can be represented as a point in an n-dimensional search space. Details about these models are presented bellow.

Genetic Algorithms

Genetic algorithms (GAs) are stochastic search strategies inspired from natural selection an evolution [1]. GAs represent admissible solutions as chromosomes, which encode the solution using genes and their values. A GA acts over a population of chromosomes using genetic operators such as *selection, crossover* or *mutation*. Parent chromosomes are selected to support crossover operations based on the values of their fitness functions and special mechanisms, such as the roulette rule algorithm [1]. Crossovers and mutations are a source of evolution induced by changing the structure of offspring chromosomes. The basic scheme of the GA is described in Box 1.

Box 1 – The genetic algorithm.

1. Initialize current population Pop={$\mathbf{X}_i$ ={x_{ij}}, i=1…NP, j=1…NG}, NP – number of chromosomes; NG-number of genes in a chromosome.
2. Compute fitness functions for the chromosomes in the current population FF_i, i=1…NP.
3. Select parent chromosomes based on fitness values and the roulette rule. $Parents$={$\mathbf{X}^*_i$, i=1…NX}, where NX - number of parent chromosomes.
4. Crossover: use parent chromosomes $\mathbf{X}^*_i$ and $\mathbf{X}^*_j$ and a random number r. If $r \le r_c$ apply crossover of parents $\mathbf{X}^*_i$ and $\mathbf{X}^*_j$ to generate offsprings $\mathbf{O}_i$ and $\mathbf{O}_j$. If $r > r_c$ parents are copied to the next generation ($\mathbf{O}_i \leftarrow \mathbf{X}^*_i$ and $\mathbf{O}_j \leftarrow \mathbf{X}^*_j$).
5. Mutation: generate a random number r. If $r \le r_m$ select a gene from the offspring chromosome and apply mutation. Elsewhere, the offspring chromosome remains unchanged.
6. Next generation: go to step 2 until a maximum number of generations is reached.

Immune Algorithm

Immune algorithms (IAs) are biologically inspired techniques designed to mimic the ability of natural organisms to protect themselves against biological aggressors and to adapt to the environment [5]. Artificial immune systems are based on the concepts of antigens and antibodies. An antigen models the optimization problem itself, while an antibody describes a possible solution. Therefore, an antibody (or solution) recognizes (or resolves) an antigen (or problem). The degree in which an antibody fits an antigen is called affinity. Better antibodies, which better recognize antigens, have higher affinities. Moreover, for faster and better antigen recognition, a higher diversified antibody population is recommended. Hence, antibodies with higher dissimilarities between one another are preferred. The principles of the IA are described in Box. 2.

Box 2 – The immune algorithm.

1. Initialize current population Pop={ $\mathbf{X}_i$ ={x_{ij}}, i=1…NA, j=1…NS}; NA – number of antibodies; NS-space dimension.
2. Compute affinity functions for antibodies AFF_i, i=1…NA.
3. Generate the proliferation pool.
4. Apply crossover and mutation operators to antibodies from the proliferation pool based on affinities AFF_i and crossover or mutation rates (see steps 3- 5 from the GA in Box 1).
5. Compute affinity functions between antibodies and the antigen and the dissimilarity degrees between antibodies.
6. Select antibodies from the proliferation pool to generate the new population, based on affinities and dissimilarities.
7. Next generation: go to step 3 until a maximum number of generations is reached.

Box 3 – The evolutionary S-PSO algorithm.

1. Initialize current population Pop={ $\mathbf{X}_i$ ={x_{ij}}, i=1…NA, j=1…NS}; NA – number of particle; NS-space dimension.
2. Compute objective functions for each particle FF_i, i=1…NP.
3. Generate the proliferation pool. r clones are generated for each particle.
4. Select particles for mutation and crossover operations (the roulette rule may be applied).
5. Mutations are induced to c_0, c_1 and c_2 parameters.
6. Apply crossovers by altering the velocity of each particle according to eq. (12).
7. Compute the new position of each particle using eq. (11).
8. Next generation: go to step 2 until a maximum number of generations is reached.

Particle Swarm Optimization

Swarm intelligence describes mechanisms inspired by nature, where groups of individuals reach some objectives through cooperation. S-PSO was developed as a simulation technique of simple social systems [2]. Strictly speaking, S-PSO is a search strategy that combines local and global search to reach an optimal or near-optimal solution. The S-PSO strategy uses a population of particles or potential solutions of the problem, which change their position, according to their own experience and the experience of other particle in the swarm, using a velocity v_i^{t+1}:

$$X_i^{t+1} = X_i^t + v_i^{t+1} \qquad (11)$$

Velocity v_i^{t+1} is computed as a linear combination of its initial value v_i^t and other two randomly weighted acceleration factors: the tendency to return to its best position so far B_i^L (local optimization), and the tendency to move towards the best position of the rest of particles in the swarm B^G (global optimization):

$$v_i^{t+1} = c_0 \cdot v_i^t + c_1 \cdot r_1 \cdot \left(B_i^L - X_i^t\right) + c_2 \cdot r_2 \cdot \left(B^G - X_i^t\right) \qquad (12)$$

where : c_0, c_1 and c_2 –weighting factors, and r_1 and r_2 – random variables in the interval [0,1]. The basic scheme of the evolutionary S-PSO algorithm is described in Box 3.

Enhanced PSO Model

Previous researches [7] proposed a different version of the S-PSO algorithm, where a particle uses information about all its neighbours and not just the best one. Our researches considered an enhanced PSO model (E-PSO), where a particle changes its position based not only on the best position so far, but also on other successful positions of itself and other particles in the current population.

Considering more or less past near-best positions might influence in a great extant the behaviour of the E-PSO. Adding too high order terms in eq. (12) could transform the guided search from the S-PSO into a random one. In fact, our studies proved that a second order E-PSO model is enough to guarantee a faster convergence. As a rule, higher order models are slower. The second order model computes the velocity for the next step using the first and the second best positions at both local and global optimization levels, denoted by $B^L_{i,I}$ and $B^L_{i,II}$:

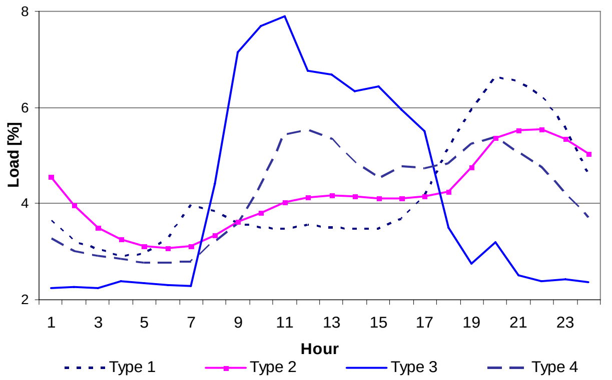

Figure 1 – Typical load profiles used to model system loads for the reactive power compensation problem.

$$\begin{aligned} v_i^{t+1} &= c_0 \cdot v_i^t + \\ &+ c_{1,I} \cdot r_{1,I} \cdot \left(B_{i,I}^L - X_i^t\right) + c_{1,II} \cdot r_{1,II} \cdot \left(B_{i,II}^L - X_i^t\right) + \\ &+ c_{2,I} \cdot r_{2,I} \cdot \left(B_I^G - X_i^t\right) + c_{2,II} \cdot r_{2,II} \cdot \left(B_{II}^G - X_i^t\right) \end{aligned} \qquad (13)$$

where $r_{1,k}$ and $r_{2,k}$ (k=I,II) are random values in [0,1].

SIMULATION RESULTS

The efficiency of the proposed E-PSO algorithm, was tested for the problem of RPC applied to a test system, which supplies a residential area in a big city in Romania. The test system has 36 nodes and 39 load sections on 8 feeders. The peak active system load was 11.966 MW, distributed among 34 MV/LV substations with rated capacities between 250 and 630 kVA and leading power factors between 0.80 and 0.90, which was equivalent to a peak reactive system load of 6.265 MVAr. System loads were modeled using TLPs for four mixed consumer types: *Type 1* - Block of apartments (10 floors) + Services; *Type 2* - Houses + Services; *Type 3* – Services; *Type 4* - Block of apartments + Houses (see Fig. 1). Reactive power compensation uses capacitor banks with rated capacity of 5 kVAr.

The simulations were driven for a period of 24 hours, during a week-day, using the algorithmic approaches described in the previous section. All approaches have used a common initial population of chromosomes, antibodies or particles, further on denoted as individuals. This population

Table 1 – Values of objective function [%] for 3 algorithms.

GA	IA	S-PSO
2.6346	2.6202	**2.6139**

Table 2 – Values of the objective function [%] for *Case I*.

Run #	1	2	3	4	5
$c_{1,I} = c_{2,I}$	0.9	0.8	0.7	0.6	0.5
$c_{1,II} = c_{2,II}$	0.1	0.2	0.3	0.4	0.5
F_{obj}	2.6150	**2.6137**	2.6185	2.6262	2.6219

Table 3 – Values of the objective function [%] for *Case II*.

Run #	1	2	3	4	5
$c_{2,I}$	1	1	1	1	1
$c_{1,I}$	1	1	1	0.6	0
$c_{2,II}$	0.8	0.5	0.5	0.5	0.5
$c_{1,II}$	0.2	0.5	0	0	0
F_{obj}	2.6152	2.6160	**2.6135**	2.6150	2.6338

comprises 10 individuals with 36 units each, the value of a unit being equal to the number of capacitor banks installed in the corresponding system node.

The efficiency of the optimization algorithms was judged in terms of the values of the objective functions from eq. (1) and of the convergence properties. For comparison, Table 1 shows the values obtained using GA, IA and S-PSO.

The E-PSO algorithm was applied for different combinations of the weighting factors $c_{1,I}$, $c_{1,II}$ and $c_{2,I}$, $c_{2,II}$ from eq (13). Thus, two base cases were considered: *Case I* – equal weights for best local and global positions of the same order ($c_{1,I} = c_{2,I}$ and $c_{1,II} = c_{2,II}$) and *Case II* – different weights for each best local or global, and first or second order position. Objective function's values for these combinations are shown in Tables 2 and 3.

The values in Table 2 suggest that it is better to use higher values for the first order local or global weighting factors. However, since the second order positions hold some information about the optimum solution, completely neglecting them could lead to worse solutions. The best solution was found for weighting factors values of 80% for the first best position and 20% for the second best position. For the case where no correlation is supposed between the weighting factors (Case II and Table 3), the global optimum deviation ($B^G_{II} - X^t_t$) from eq. (13) is the most influential second order term. Our experiments shown that the best results were obtained with higher values for the first order terms ($c_{1,I}$ and $c_{2,I}$) and lower values for the second order terms ($c_{1,II}$ and $c_{2,II}$). For instance, as is the case in Table 3, a possible combination is: $c_{1,I} = 1$, $c_{2,I} = 1$, $c_{1,II} = 0$, $c_{2,II} = 0.5$.

The E-PSO algorithm behaves better not only in terms of the value of the objective function (see the bold values from Tables 1 – 3), but also and especially in terms of the convergence properties. In this context, a comparison of the convergence properties of the three search procedures (GA, IA and S-PSO) shown that the GA behaves the worst and the S-PSO algorithm – the best. Moreover, a comparison between S-PSO and E-PSO algorithms proved that the second one is faster. These findings are supported by the graphic representation from Fig. 2.

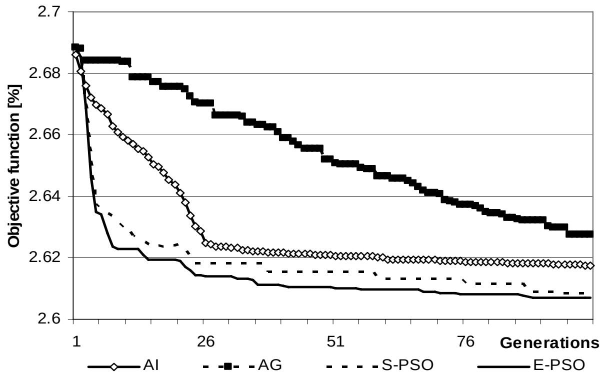

Figure 2 – Evolution of the values of the objective function for the best individuals for the four searching procedures.

CONCLUSIONS

An enhanced particle swarm optimization (E-PSO) technique has been used to approach the problem of reactive power compensation in distribution systems. The optimization problem uses a multiobjective formulation based on mimization of energy losses and voltage deviations, subject to voltage and compensation restrictions. The E-PSO algorithm uses second order correction term for the particles' velocities. The performance of the new approach was demonstrated using simulations on a 36 nodes radial test system.

REFERENCES

[1] T. Back, 1996, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, New York, USA.

[2] A.A. Esmin, G. Lambert-Torres,A.C. Zambroni de Souza, 2005, "A hybrid particle swarm optimization applied to loss power minimization", *Power Systems, IEEE Transactions on*, Vol. 20, Issue 2, 859 – 866.

[3] L. Furong, A. Chebbo, R.K. Aggarwal, 2004, "Genetic algorithms for optimal reactive power compensation on the national grid system", *Power Systems, IEEE Transactions on*, Vol. 20, Issue 1, 493 – 500.

[4] M. Gavrilas, C.V. Sfintes, 2006, Fuzzy Self-Organizing Approach to Electric Load Profiling, *Bul. Instit. Politehnic Iasi, Tom LVI, Fasc.*, 1181-1186.

[5] S. Ishak, A.F. Abidin; T.K.A. Rahman, 2004, "Static Var compensator planning using artificial immune system for loss minimization and voltage improvement", *Proc. Power and Energy Conf.*, 41 - 45

[6] W.-S Jwo,. C.-C Liu, Y.-T Hsiao, 1995, „Hybrid expert system and simulated annealing approach to optimal reactive power planning", *Generation, Transmission and Distribution, IEE Proceed.*, vol. 142, 381 – 385.

[7] R. Mendes, J. Kennedy, 2004, The Fully Informed Particle Swarm: Simpler, Maybe Better, *Evolutionary Computation, IEEE Trans. on*, Vol.3, 201-211.