Radial Distribution Network Topology Optimization Using Genetic Algorithms Considering Uncertain Load and Distributed Generation

Vitor Hugo Ribeiro
Electrical Engineering Department
School of Engineering, Polytechnic Institute of Porto
Porto, Portugal
vitorhmribeiro@gmail.com

Filipe Azevedo
Electrical Engineering Department
INESC TEC and School of Engineering, Polytechnic Institute of Porto
Porto, Portugal
filipe.azevedo@inescporto.pt

***Abstract*— This paper aims to study distribution network topology optimization considering uncertain load and distributed generation. Gradual increase of distributed generation in distribution network leads the network operator companies to concern more about having the best network topology, so their costs can be the lowest. MATLAB™ genetic algorithms function is used to model this mathematical problem in its basic definition. A stochastic multi-objective programming algorithm is implemented and a decision maker applied to choose the best solution of non-dominated solutions set found.**

Keywords—Topology Optimization; Genetic Algorithms; Distributed Generation; Uncertainties Modeling; Power Loss Reduction; Reliability; Voltage Levels; Robustness.

I. INTRODUCTION

The operation of distribution networks is made, usually, with radial topology. However, to ensure reliability index, the network owns other branches, usually not active, so it can ensure feed of a partial number of loads in case of failure occurrence [1].

The regular topology is obtained by setting the branches that need to be active so the network losses, reliability index and power quality are the best. Switches are placed in the key bus of the network. Changing their operating state in a coordinated manner, between on and off, makes it possible to change the network topology maintaining the radiality [1].

The inclusion in networks of distributed generation (DG) technologies represents a larger proximity between generation and consumption centers. This fact leads itself to the decrease of power losses value. The network reconfiguration can be responsible for promoting more efficient use of distributed generators by analyzing the availability of each source for injecting energy into the system. It is also interesting because it allows the grid to operate with lower resistive losses or transmit more power from DG than in static configurations by dynamically changing its topology [1].

There are different technologies of distributed generation. For this paper only three are addressed: solar, wind and cogeneration power. Due to unpredictable characteristics of primary sources (solar and wind power), the distribution network, where these may be included, is subjected to generation fluctuations at the specific buses. This fact result in different branch power flow values, causing an hypothetical configuration that would firstly correspond to best network topology to have worst performance, when compared with other viable configurations.

In reconfiguration of distribution network topology problem it is necessary to avoid technical constraint violations of several parameters so that network operation allows the power delivery with necessary quality. In particular, imposed voltage limits and capability limits of network facilities and devices as branches and substations.

Power losses and voltage drop along the network are important factors to economic-commercial balance of the concessionaires, since they contribute to compose energy supply rates.

There are different goals to the distribution network topology optimization relying mainly on point of view. If to the consumers the main concern is the maximization of reliability and power quality, to the companies the main goal is the profit obtained by having minimum power losses and maximum number of customers connected to the network.

In order to represent these interests, the optimal topology in this paper is represented by the solution presenting the best ratio between network real power losses (RPL), reliability of the system and voltage values proximity to the ideal at all load buses. The best ratio depends always on the weight set to each of the parameter above. The details can be found in section V. Thus, this concept fits in a multi-objective problem feature wherein the solutions are evaluated by set of fitness functions [2].

It is a common sense that, having the best network topology at all times is a real challenge. Either because the problem is of combinatorial nature, and so it is computationally complex, or because the technical component which would require the operator companies to have a wide number of network cut and sectioning equipment, so they allow these operations to be both automatically and instantaneous. Several researches have their main goal to real-time reconfiguration problem considering DG, associating the smart grids concept [1, 3].

In fact, while in most cases it is not possible to change network topology in every network buses in such short periods it is necessary to think of a best solution concept that would be independent from the load and DG fluctuations. This concept is not feasible to all load and DG values. Thus, becomes important to think of a solution that would be the best taking inherent uncertainties in consideration to a specified interval, fitting this approach to the distribution topology planning problem.

The association of uncertainties to the multi-objective topology optimization makes the problem even more complex. It leads to an increase in the amount of solution evaluation processes and largely increases the complexity of making a decision between solutions which, by hypothesis, correspond to non-dominated solutions.

This concept has been studied by researchers, presenting in literature some ways of dealing with uncertainties. Modelling and transposing uncertainty concept into search algorithms towards optimum solution of multi-objective problems. In this paper, the representation of the different DG is detailed in section III.

A. Problem optimization model characteristics

There are different manners to model this problem based on the optimization variables definition. The classical problem variables are binary variables, representing state of branches or switches in the network. Reference [1] proposes a real-coded algorithm which consists in an encoder of binary variables representing them as real variables.

Network reconfiguration problem was treated as an integer programming problem with linear and nonlinear constraints and all decision variables restricted to be integers (considered as binary variables). The general mathematical model is represented by (1).

$$\begin{aligned} &\min \quad f(x) \\ &\quad g_j(x) \le b_j, && j=1,\dots,r_1 \\ &\quad h_j(x) \le b_j && j=1,\dots,r_1 \\ &\quad x_i^L \le x_i \le x_i^U && \text{integer},\ i=1,\dots,n_1 \\ &\quad x=[x_1,x_2,\dots,x_{n1}]^T \end{aligned} \tag{1}$$

Several classical computation techniques have been proposed in literature for solving integer and mixed integer programming problems (MIPP), such as branch and bound technique, cutting planes technique, outer approximation technique, etc [1].

Many of the proposed techniques are based in stochastic algorithms and adapted for mixed integer programming problems. Traditional techniques as Simulated Annealing, Differential Evolution and Particle Swarm Optimization are used to solve MIPP [1].

II. Genetic Algorithms Processes

Genetic algorithms are general purpose population based stochastic search techniques which mimic the principles of natural selection and genetics laid down by Charles Darwin. The concept of Genetic Algorithm (GA) was introduced by Holland. This approach was first used to solve optimization problem by De-Jong. It mainly consists in a population of individuals transformed by three genetic operators: Selection, Crossover and Mutation [4-6].

A. Detailed operator steps

Selection operator creates a new population (or generation) by selecting individuals from the old population, biased towards the best. This operator can be implemented in a variety of ways, although in the proposed methodology a technique known as Stochastic Tournament is used. Every time we want to select an individual for reproduction, we choose two, at random, and the best wins with some fixed probability, typically 0.8 [6, 7].

Crossover is the main genetic operator and the engine of genetic algorithms. It consists in exchanging chromosome parts between individuals. The crossover operator can be implemented by selecting a random crossover point in the chromosome, and then swapping the genes that reside between the crossover point and the end of the chromosome.

The last genetic operator is mutation and, in its simplest form, it consists in toggling a random bit in an individual. The selection and crossover do not introduce any new genetic material in the population. Thus, the mutation operator is used in order to guarantee the possibility of searching in any particular subspace of the problem space, preventing the search of finishing in a local optimum [6, 7, 9].

Similarly to other stochastic methods, GA has a certain number of parameters, such as population size, probabilities related to genetic operators and number of individuals in the tournament, that need to be selected with care, since the performance of a GA depends heavily on these values [6, 7,9].

B. MATLAB Genetic Algorithm

The need of using an optimization technique, and being GA such a promising technique, leads the authors to use MATLAB’s genetic algorithm generic function. This software has a substantial capability to adapt at various optimization problems and has a real simple form to deal with different models. Other MATLAB’s GA function specific to deal with multi-objective problems presented some shortcomings that make its use inappropriate.

Although not a rule, it is logic to admit that this function was expectable, due to the lack of specificity, having some inconveniences. Not so expectable was the fact that later this becomes an insurmountable barrier. In fact, the support presented by MathWorks help website content is very complete and entirely useful, under didactical perspective.

As detailed in section IV this problem introduce an increased difficulty in what the definition of constraints concerns, mostly at its implementation.

One of the both most important and hard implementation constraints in this generic tool is radiality. It can be granted using different approaches, but defining it as constraint in GA MATLAB's function is, in the authors' perspective, impossible. Because the constraints input topology is proper for linear constraints.

Literature presents different manners to deal with general network radiality constraint. In association with MATPOWER software it is very simple to use the architecture presented by [13]. Whose authors refers two theoretical conditions that maintain the network in a radial configuration: (1) the solution must have one less branch then buses; (2) the solution must be connected. The second condition can be hard to implement. There are many heuristic based algorithms referred in literature to check network connectivity. MATPOWER simplifies having only two functions used to test the existence of both islands and disconnected bus [13].

It is logical to consider that if GA generated populations are radial, the search of the algorithm is more restrict. So, if a method of controlling the GA solutions generation through reproduction processes exists, the problem would have much lower computational and mathematical weights.

Reference [8] puts constraint handling methods used in classical optimization algorithms into two groups: (1) generic methods that do not exploit the mathematical of the constraint, and (2) specific methods that are only applicable to a special type of constraints.

Generic methods have an easier implementation. In this, penalty functions can be implemented inside GA fitness function or other method related fitness evaluation function. The solutions can be feasible or not feasible depending on if it respects or not the set of the constraints that we attempt to respect.

The number of causes and the non-compliance degree can take the penalty function to assume three different approaches: multi-level penalty functions, dynamic penalty functions, and penalty functions involving temperature-based evolution of penalty parameters with repair operators [8].

Reference [8] relates another approach to control the GA search in nonlinear programing problem which mainly acts in the selection function. Using tournament selection with two individuals: (1) a feasible solution is always selected when compared with an unfeasible one; (2) comparing unfeasible solutions the one with less constraint violation is selected; (3) comparing feasible solutions the one with best fitness value is selected with a given probability.

Unfortunately, MATLAB's GA function applied to integer-constrained problems prevents the user to change or even to create any selection function. In this work a multi-level penalty function is created to match different fitness values for solutions, which do not respect the different requirements of a solution to be radial. Although having a selection technique for unfeasible solutions control, it is not possible to control unfeasible solutions from radiality constraint violation using the same selection process.

III. MODELING UNCERTAINTIES OF DISTRIBUTED GENERATION AND LOADS

The three DG technologies addressed are: wind, solar photovoltaic and cogeneration power. Because of its unpredictability it is necessary for their evaluation to model each different form of technology. This modeling happens through a probabilistic analysis in which a controlled random number is generated for each technology and addressed to the buses including them.

Since the technologies rely on different sources of energy, modeling DG uncertainties will have to represent each of them separately.

Cogeneration is not dependent on a source with uncertain primary energy. Then, logically, uncertain generation is associated only to the operating point of the power generation machine. It is assumed that this type of technology is targeted to industries and for these it is quite a compensatory use. So it is rational to assume that they will have a relatively constant production. Based on this, it is considered that output value will always be close to the nominal. The uncertainty interval is defined between 80% and 100% of the generator installed capacity.

Solar energy shows typical patterns of variation. Although analyzing equal time periods of consecutive days, these patterns can take quite disparate values due to weather conditions variation. These standards are related to a daily analysis. Although the weather variations are often sudden and unexpected, in a daytime period, available irradiation morphologically follows a normal distribution as in (2). The parameters mean and standard deviation are stated as 0 and 1, respectively. Normal distribution used is truncated to match the percentage of used installed capacity at each DG point. In this model it is disregarded the change in absolute value for an annual review, the conversion efficiency of the panels and other conditioner factors as the influence of shadows.

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{\frac{-(x-\mu)^2}{2\sigma^2}}\ ,-\infty < x < \infty \qquad (2)$$

Wind power is usually represented by a Weibull distribution. This distribution has two parameters associated, *c* and *k* that can be changed to adapt to different mathematical cases. The variable *k* corresponds to the shape parameter and the variable *c* to the scale parameter. In this paper for modeling wind power was used *c=12 m/s* and *k=2*. In addition, to wind power representation it is crucial to involve mechanic characteristic of eolic turbine. Even though this depends on each technology, but typically follows the representation present in Figure (1).

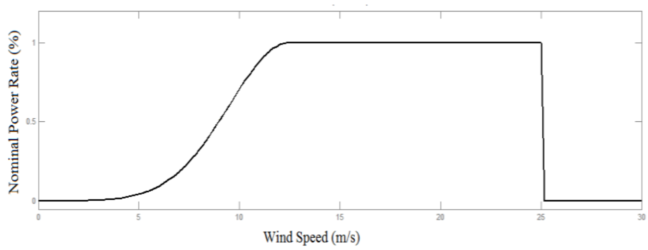

Fig. 1. Mechanical characteristic of typical wind power turbine

MATLAB software already includes Weibull and Normal Distributions functions. With the generation of three different random numbers, DG power are modeled and, from there, power flow calculation runs in order to obtain the network RPL value.

In cogeneration power approach, the random number is crossed with an affine function that represents the operating point of the generator. In photovoltaic modeling the same random number is biased to match normal distribution. Finally, in wind power case this representation is bolder. The random number, in addition of being biased to match Weibull distribution for obtaining primary energy value, is then considered affected by the mechanical characteristic of the Eolic turbine.

Hence, the installed capacity percentage values used for the analysis of a static occurrence are obtained. Considering that these values correspond to a single iteration, they will contribute to uncertainty creation in fitness values, in particular, system RPL and voltage values at the buses.

The suggested representation of load uncertainties follows the same principles of cogeneration modeling method. Wherein vary with a uniform distribution between 50% and 100% of each load value, present in the case definition.

IV. Proposed Matlab algorithm

The proposed algorithm for the problem resolution is divided in three stages: search, simulation and decision. Genetic Algorithms are used to search the set of non-dominated radial solutions based on three parameters: real power loss, reliability index of the entire system and node voltage module. The algorithm is based in a mono-objective convergence towards topology presenting best system real power loss without considering DG penetration.

After the set of non-dominated solutions, called Pareto Set, be found, the DG penetration based in section III is modeled. The Pareto Set solutions fitness is reevaluated and the individuals first found by GA have now uncertain fitness, in particular power losses and voltage at nodes values.

The decision counting on these three parameters is made with weighted average percentage of the difference between each stage value and best value in the set for each parameter.

The algorithm is a result of the aggregation of GA MATLAB's function that contemplates the principal search of genetic algorithms. This function is adapted to the problem so there is no violation of any characteristic of binary variables and non-linear restrictions can be implemented. The function presents panoply of possibilities of matching the variations of canonical genetic algorithms and aims towards the adaptability to extremely different problem features.

The specific parameters defined in GA function of the proposed algorithm are shown in [7].

Given the specific problem features, shown in formulation section, and assuming that it is only possible to define linear restrictions in integer optimization with MATLAB™ GA function with topology: $A \cdot x \leq b$, it is impracticable trying to restrict non radial solutions at basic processes of generation of solutions. The option is allowing the non-admissible solutions to be generated controlling its spread inside fitness function. [7]

Adopting this control topology, it is rapidly understandable that a large number of not admissible solutions will be generated by GA process passing through all GA steps. Therefore, the own features of binary variables problem optimization allied with basic processes of genetic algorithms, make that two admissible solutions affected with crossover or mutation techniques, generate new individuals in non-admissible region and logically with really poor performance. This happens mainly because of the active branch limit restriction.

The used genetic algorithm function of MATLAB is based on a Laplace crossover, power mutation, tournament selection technique, a truncation procedure for integer restrictions and a constraint handling approach [1, 4, 5, 10].

There are, for this algorithm, six computational steps detailed in [1]. Based on this concept, and assuming that radiality constraints need, by imposition of GA MATLAB™ function, to be controlled inside fitness function, three steps of control were defined in it.

For a solution to be radial it is necessary to exist only $n\_bus - 1$ number of active branches, no isolated bus and no islands on the topology modeled by solution being evaluated.

Testing the number of islands and the number of isolated buses is really simple in association with MATPOWER, software developed by Ray Zimmerman, used also in the proposed algorithm defined in Figure (2).

Besides computational speed, other questions associated to the basics of the difficulty arise: How to evaluate a solution not evaluable? How to calculate real power losses value if there is any isolated bus?

The primary evaluation of the solution used for GA convergence is made by these three parameters and a ranking is assigned by fitness value to each solution that does not pass this test phase. A control variable, named *VarCtr*, is created and the ranking is made based on the design of this control variable as shown in Figure (3).

In order to deal with unfeasible solutions, a control stage of impractible solutions inside the fitness function was made. A penalty is set and matches a major fitness value when the solution is not in admissible region. Considering the existence of non-admissible solutions with more or less proximity to the admissible region, a tree with different control stages and penalty values was designed as shown in Figure (3). Note that it is not the most efficient manner to solve this problem. This should be done as closely as possible to the basic definition of generation of new populations of individuals.

The parameters are: network isolated bus number, network active branches number and network islands number. The second referred parameter is controlled also by MATLAB GA function entrance variables, but the function has some difficulties working with constrained and bounded search [7].

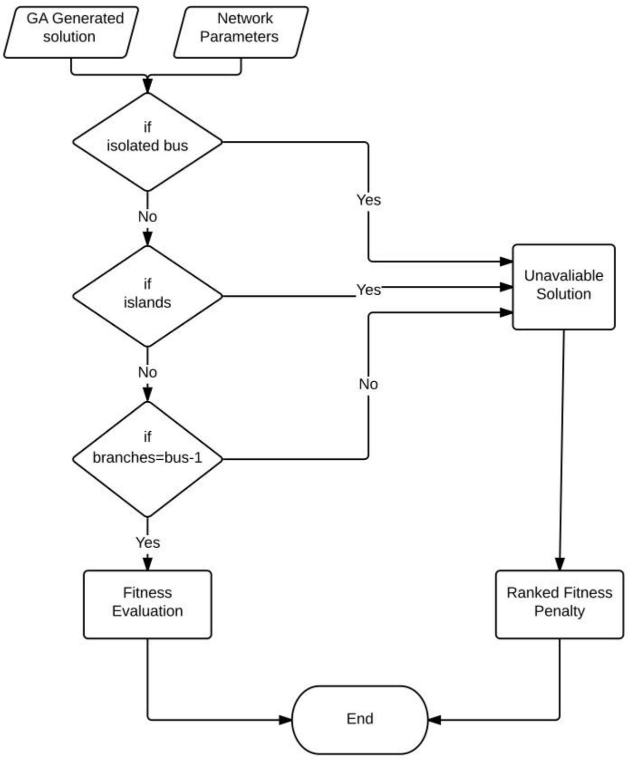

Fig. 2. Solution evaluation steps

If the number of active branches, represented by the solution being evaluated, is inferior to the initially set value for the case (depending on the network), no solution is admissible. Then this is the major penalty of a solution. The penalty is bigger if the solution is checking fewer parameters, as shown in Figure (3).

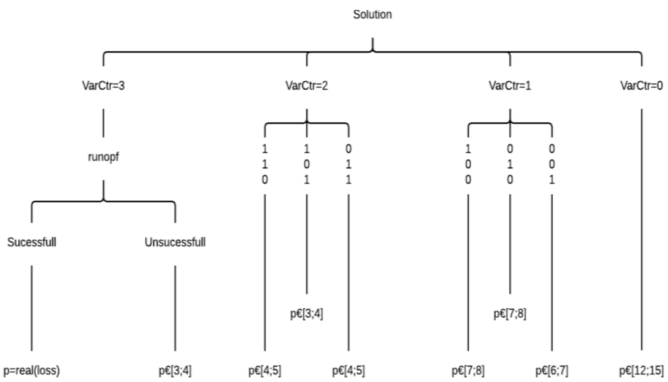

Fig. 3. Penalty function implemented architecture inside fitness function

However, the difficulty of a robust structure design and definition, so these penalties are directly proportional to the distance of the solution to admissible region has major significance. A hypothetic solution that does not verify any evaluation parameters can be truly close to both feasible or even the optimal of the problem.

The idea of creating an algorithm who recognizes the patterns of all admissible solutions for any case and calculates the ideal distance of an admissible solution, defining the real quality of each individual by its fitness so the GA can have the best thinkable convergence, appears to have a considerably colossal impracticable component. It is comprehensible that these must be penalized. The patterns need to be worked closest to the ideal, and genetic algorithms can make this pattern to cross generations by the recombination of individuals.

V. Decision Maker

As stated at previoussection, after obtaining Pareto Set by mono-objective GA, all solutions have their fitness reevaluated according with DG values defined in the case study. Four parameters to evaluate each solution are considered: real power losses, system reliability, voltage difference and robustness.

Considering uncertainties of load profiles and DG values leads both real power losses and voltage difference to have more than one fitness value. Being network power losses the main considered parameter, robustness concept is included to represent the range of each solution power losses.

The number of real power losses and voltage difference values rely on how much iterations are considered. A single iteration represents a single state of loads and DG power in each bus. Decision maker algorithm is made in order to convert all of the uncertain fitness values into only one parameter.

A. Process steps

After all evaluation values are found it is necessary to deal with them. Power losses and voltage at buses are converted to match a single iteration.

In the final decision all four evaluation parameters are represented by ratio corresponding to the division of each approached value by the minimum found in Pareto Set in the same category. Power losses and voltage difference approached values to each solution of the set are represented by the correspondent average value of all iterations.

System reliability parameter is not affected by the consideration of uncertainties in loads and DG. Hence, there is no need to consider any approach. A solution is better as higher is reliability. Then, in its calculation it is necessary to assume the value as negative. Its ratio is represented dividing each value by the optimal reliability found in Pareto Set.

The process to obtain robustness parameter of each solution comprises three steps: (1) Initially, the RPL value is obtained; (1) therefore, the sum of the difference between power flow obtained and the ideal, 1 *p.u.*; (3) lastly, robustness ratio is obtained dividing this difference by the optimal best.

Each solution has only one fitness value and it is necessary to consider the influence of each parameter expressed in (3).

$$final\ fitness = w_1 \cdot r_1 + w_2 \cdot r_2 + w_3 \cdot r_3 + w_4 \cdot r_4 \qquad (3)$$

Where, $w_{[1,\ 5}$ corresponds to the values of the weights assigned to each of the parameters and $r_{[1,\ 4]}$ corresponds to the values of each parameter ratios of the evaluation function.

Hence, final fitness values of each solution are obtained. The best solution is always the one having the lowest final fitness value.

VI. CASE STUDY

In this section was used the 33 buses network described in [10]. Twenty simulations were performed and solutions are presented in Table I. An initial solution is considered as an input parameter of GA MATLAB function is analyzed and the can itself find a solution with better fitness value.

TABLE I. RESUTLS

Number of iterations with the same solution	**Topologies found**
10	6,10,35,36,37
5	6,8,10,36,37
3	6,10,14,17,37
1	11,28,32,33,34
1	6,10,28,35,36

In Table I, we can observe that GA could not find any better solution during 20 simulations. In GA output graphs obtained from each simulation all of the issues, detailed in previous sections, occurs. These issues are related to penalty function implementation. Figure (4) represents a typical convergence of GA without considering any initial solution. We can perceive when the algorithm find a unique feasible solution in late generations, GA overemphasize this sole feasible solution. This results in prematurely convergence near this solution.

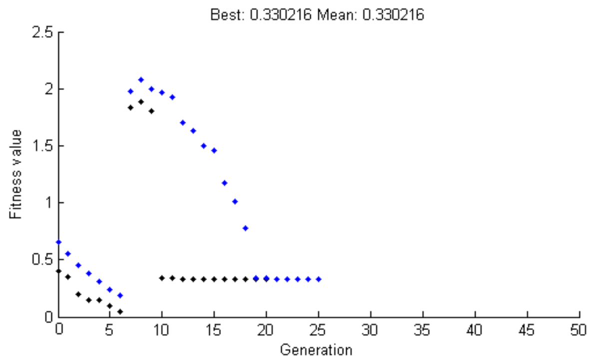

Fig. 4. Example of unsuccessful convergence

The best solution found by mono-objective GA do not correspond to the best one found by other authors doesn't corresponding to the global optimum of the problem.

VII. CONCLUSION

In this paper was addressed the influence of load and distributed generation uncertainty in the optimization problem of the optimal distribution network topology. Was used the generic optimization tool of MATLAB™, GA MATLAB function. It is quite difficult to introduce concepts approached in the literature which aims at implementing improvements to this canonical approach. Thus, it was impossible to control and restrict the number of non-feasible solutions to generate solutions by the algorithm as would be desired.

The problems found using this approach are mainly associated with the GA search. This happen especially because the option taken to represent the fitness value of non-dominated solutions was using the penalty function.

Is concluded that the traditional binary variables representation leads to a lack of robustness in GA search under viable solutions. Is currently being developed, by the authors, a new and specific genetic algorithm that is avoiding the excessive number of unfeasible solutions using different concepts and methods addressed in the literature. Thus, it is intended that future addressing of this problem become more robust and especially allowing manipulation of GA search process.

REFERENCES

[1] D.P. Bernardon, A.P.C. Mello, L.L. Pfitscher, L.N. Canha, A.R. Abaide and A.A.B. Ferreira, "Real-time reconfiguration of distribution network with distributed generation", Electric Power Systems Research, 107 (2014), 59– 67

[2] V. Farahani, S. H. H. Sadeghi, "An Improved Reconfiguration Method for Maximum Loss Reduction Using Discrete Genetic Algorithm", The 4th International Power Engineering and Optimization Conf. (PEOCO2010), Shah Alam, Selangor, Malaysia: 23-24 June 2010

[3] Colleen Lueken, Pedro M. S. Crvalho and Jay Apt, "Distribution grid reconfiguration reduces power losses and helps integrate renewables", Energy Policy 48 (2012) 260–273

[4] Kusum Deep, Manoj Thakur, "A new mutation operator for real coded genetic algorithms", Applied Mathematics and Computation 193 (2007), 211–230

[5] Kusum Deep, Manoj Thakur, "A new crossover operator for real coded genetic algorithms", Applied Mathematics and Computation 188 (2007), 895–911

[6] Vladimiro Miranda, J. V. Ranito, L. M. Proença, "Genetic Algorithms in Optimal Multistage Distribution Network Planning", IEEE Transactions on Power Systems Vol. 9, No. 4, pp. 1927-1933, November 1994.

[7] Eduardo G. Carrano, Luiz A. E. Soares, Ricardo H. C. Takahashi and Oriane M. Neto," Electric Distribution Network Multiobjective Design Using a Problem-Specific Genetic Algorithm", IEEE Transactions on Power Delivery , Vol. 1, No. 2, April, 2006

[8] Kalyanmoy Deb, "An efficient constraint handling method for genetic algorithms", Comput. Methods Appl. Mech. Engrg. 186 (2000), 311-338

[9] Marina Lavorato, John F. Franco, Marcos J. Rider, and Rubén Romero, "Imposing Radiality Constraints in Distribution System Optimization Problems", IEEE Transactions on Power Systems, Vol. 27, No. 1, February 2012

[10] Yuan-Kang Wu, Member, IEEE, Ching-Yin Lee, Le-Chang Liu, and Shao-Hong Tsai, "Study of Reconfiguration for the Distribution System With Distributed Generators", IEEE Transactions on Power Delivery, Vol. 25, No. 3, July 2010