CrossMark

Reconfiguration of Radial Distribution Systems with Variable Demands Using the Clonal Selection Algorithm and the Specialized Genetic Algorithm of Chu–Beasley

Simone S. F. Souza[1] · Ruben Romero[1] · Jorge Pereira[2] · João T. Saraiva[2]

Abstract This paper presents two new approaches to solve the reconfiguration problem of electrical distribution systems (EDSs) with variable demands, using the CLONALG and the SGACB algorithms. The CLONALG is a combinatorial optimization technique inspired by biological immune systems, which aims at reproducing the main properties and functions of the system. The SGACB is an optimization algorithm inspired by natural selection and the evolution of species. The reconfiguration problem with variable demands is a complex combinatorial problem that aims at identifying the best radial topology for an EDS, while satisfying all technical constraints at every demand level and minimizing the cost of energy losses in a given operation period. Both algorithms were implemented in C++ and test systems with 33, 84, and 136 nodes, as well as a real system with 417 nodes, in order to validate the proposed methods. The obtained results were compared with results available in the literature in order to verify the efficiency of the proposed approaches.

Keywords Distribution systems reconfiguration · Variable demands · Clonal selection algorithm · Specialized genetic

algorithm of Chu–Beasley · Artificial immune systems · Metaheuristics

1 Introduction

In recent years, electrical distribution systems (EDSs) have been the target of large investments and research toward modernizing and improving the automation of the operation of these systems, so as to increase the reliability, efficiency, and security of the networks. In this context, one available mechanism to improve the efficiency of EDS operation is distribution system reconfiguration (DSR), which is a research area that has been frequently addressed by the scientific community in recent years.

The DSR is a complex combinatorial problem that can be modeled as a mixed-integer nonlinear programming (MINLP) problem (Abdelaziz et al. 2009). The main objective of the DSR problem generally is the minimization of active power losses in the EDS. In this case, the objective function is subject to constraints related to the system's operation, including nodal voltage limits, branch current capacity limits, the first and second law of Kirchhoff (active and reactive power balance), and the radial operation of the system. Apart from this typical objective, the DSR problem can also be solved with variable demands, where the objective is minimizing the cost of the power losses along an operation period. Due the combinatorial nature, the DSR problem is a problem which is difficult to solve using exact methods, as there are many combinations of solutions. Thus, an alternative to solve this problem, is the use of the intelligent methods such as heuristics, metaheuristics, artificial neural networks, and artificial immune systems.

In this sense, in the literature is possible found many papers about the DSR problem, but the most of the papers

✉ Simone S. F. Souza
simonefrutuoso.mat@gmail.com

Ruben Romero
ruben@dee.feis.unesp.br

Jorge Pereira
jpereira@inesctec.pt

João T. Saraiva
jsaraiva@fe.up.pt

[1] Electrical Engineering Department, Unesp, Univ Estadual Paulista, Av. Brasil 56, Ilha Solteira, SP 15385-000, Brazil

[2] INESC TEC and University of Porto, Av. Dr. Roberto Frias 378, Porto 4200-465, Portugal

just consider fixed demand in the analysis. In the mid-1970s, Merlin and Back (1975) proposed one of the first heuristic algorithms to solve the DSR problem. Similar but more sophisticated approaches were proposed in Baran and Wu (1989) and Civanlar et al. (1988). These approaches are good, but not satisfactory to large problems. The metaheuristics such as genetic algorithms (Guimarães et al. 2010; Nara et al. 1992; Souza et al. 2015a), simulated annealing (Chang and Kuo 1994), Tabu search (Zhang et al. 2007; Abdelaziz et al. 2010), ant colony systems (Carpaneto and Chicco 2008), and particle swarm optimization (Abdelaziz et al. 2009) are more efficient. The application of other approaches, such as artificial neural networks (Kim et al. 1993; Salazar et al. 2006) and artificial immune systems (Kavousi-Fard and Niknam 2014; Souza et al. 2015b), can also be found in the literature. With regard to the DSR problem with variable demands, there are fewer reports and publications available in the literature. In Bueno et al. (2004), the authors propose the use of branch exchange by energy. Reference Queiroz and Lyra (2009) presents a hybrid genetic algorithm to solve the problem with variable demands. On the other hand, Inoue et al. (2014) propose a novel optimization method with compressed search space using a binary decision diagram and Possagnolo 2015 details four variations of the variable neighborhood search (VNS).

In this paper is proposed the application of two methods to solve the DSR problem with variable demands: the clonal selection algorithm (CLONALG) (de Castro and Von Zuben 2000) and the specialized genetic algorithm of Chu–Beasley (SGACB) (Silva et al. 2005). Both algorithms are efficient optimization techniques. The CLONALG is inspired by the functioning of biological immune systems, providing a computational emulation of the main properties and functionalities of the organism. The SGACB is an optimization technique inspired by natural selection and the evolution of species. In order to evaluate the quality of each proposed solution, a specialized sweep load flow for radial EDS is used (Shirmohammadi et al. 1988) for each demand level to be analyzed as a way of calculating the power losses. The proposed methods were evaluated using test systems with 33, 84, and 136 nodes, along with a real system with 417 nodes. The obtained results were compared to the results available in the specialized literature in order to assess the efficiency of the proposed methods.

It is noteworthy that the main novelty/contribution of this paper is the two new proposed approaches to solve the DSR problem with variable demands and the method used to obtain the variable demands making the analysis nearest of a real situation.

2 Mathematical Model for the DSR Problem with Variable Demands

The DSR problem with variable demands, considering a single radial configuration and a symmetrical and balanced system, can be modeled as an MINLP problem with the structure shown in (1)–(12), as presented in Franco et al. (2012):

$$\min v = \sum_{d \in \Omega_d} \sum_{ij \in \Omega_l} c_d^l \Delta_d$$
$$[g_{ij}x_{ij}(V_{i,d}^2 + V_{j,d}^2 - 2V_{i,d}V_{j,d}\cos\theta_{ij,d})]^{\text{s.t.}} \quad (1)$$

$$Ps_{i,d} - Pd_{i,d} - \sum_{j \in \Omega_{bi}} (x_{ij}P_{ij,d})$$
$$= 0 \quad \forall i \in \Omega_b, \forall d \in \Omega_d \quad (2)$$

$$Qs_{i,d} - Qd_{i,d} - \sum_{j \in \Omega_{bi}} (x_{ij}Q_{ij,d})$$
$$= 0 \quad \forall i \in \Omega_b, \forall d \in \Omega_d \quad (3)$$

$$\underline{V} \le V_{i,d} \le \overline{V} \quad \forall i \in \Omega_b, \forall d \in \Omega_d \quad (4)$$

$$(P_{ij,d}^2 + Q_{ij,d}^2) \le \overline{S_{ij,d}^2}x_{ij} \quad \forall ij \in \Omega_l, \forall d \in \Omega_d \quad (5)$$

$$(I_{rij,d}^2 + I_{mij,d}^2) \le \overline{I_{ij,d}^2}x_{ij} \quad \forall ij \in \Omega_l, \forall d \in \Omega_d \quad (6)$$

$$x_{ij} \in \{0, 1\} \quad \forall ij \in \Omega_l \quad (7)$$

$$\sum_{(ij) \in \Omega_l} x_{ij} = n_b - 1 \quad (8)$$

In this formulation: Ω_l is the set of circuits; Ω_b is the set of nodes; Ω_{bi} is the set of nodes connected at node i; Ω_d is the set of demands; c_d^l is the cost of energy at demand level d; Δ_d is the duration of demand level d; g_{ij} is the conductance of circuit ij; $V_{i,d}$ is the voltage magnitude at node i at demand level d; $\theta_{ij,d}$ is the phase angle difference between nodes i and j at demand level d; b_{ij} is the susceptance of circuit ij; $P_{ij,d}$ is the active power flow that goes from node i to node j at demand level d; $Q_{ij,d}$ is the reactive power flow that goes from node i to node j at demand level d; $Ps_{i,d}$ is the active power supplied by the substation at node i at demand level d; $Qs_{i,d}$ is the reactive power supplied by the substation at node i at demand level d; $Pd_{i,d}$ is the active power demanded at node i at demand level d; $Qd_{i,d}$ is the reactive power demanded at node i at demand level d; $\underline{V}$ is the minimum voltage magnitude; $\overline{V}$ is the maximum voltage magnitude; $\overline{S_{ij,d}}$ is the maximum apparent power of circuit ij at demand level d; n_b is the number of nodes in the system; and x_{ij} is the binary decision variable that represents the state (connected or disconnected) of circuit ij.

Equation (1) represents the objective function of the DSR problem with variable demands, where "v" corresponds to the cost of energy losses in the EDS to be minimized. The mathematical model considers physical constraints, the specifications of system components, and operational conditions.

Constraints (2) and (3) represent active and the reactive nodal balance equations in which the active and reactive power flows $P_{ij,d}$ and $Q_{ij,d}$ are calculated, respectively, using (9) and (10).

$$P_{ij,d} = V_{i,d}^2 g_{ij} - V_{i,d}V_{j,d}(g_{ij}\cos\theta_{ij,d} + b_{ij}sen\theta_{ij,d}) \quad (9)$$
$$Q_{ij,d} = -V_{i,d}^2 b_{ij} - V_{i,d}V_{j,d}(g_{ij}sen\theta_{ij,d} - b_{ij}\cos\theta_{ij,d}) \quad (10)$$

Constraint (4) represents the voltage magnitude limits for each node of the EDS, as defined by regulatory standards. The power flow in each circuit is limited by (5). And the constraint of the current limits is express by (6), in which the real current and imaginary current $I_{rij,d}$ and $I_{mij,d}$ are calculated, respectively, using (11) and (12).

$$I_{rij,d} = G_{ij}(V_{i,d}\cos\theta_{i,d} - V_{i,d}\cos\theta_{j,d})$$
$$- B_{ij}(V_{i,d}\sin\theta_{i,d} - V_{i,d}\sin\theta_{j,d}) \quad (11)$$
$$I_{rij,d} = G_{ij}(V_{i,d}\sin\theta_{i,d} - V_{i,d}\sin\theta_{j,d})$$
$$+ B_{ij}(V_{i,d}\cos\theta_{i,d} - V_{i,d}\cos\theta_{j,d}) \quad (12)$$

Constraint (7) corresponds to the binary nature of the decision variables, according to which x_{ij} can take two values as follows: When it is 0 (zero), circuit ij is open (or disconnected), and when it is 1 (one), circuit ij is closed (or connected).

Constraint (8) presents one of the necessary conditions to guarantee the radial operation of the EDS, namely, that a solution to the problem must have $(nb - 1)$ active circuits. The other necessary condition is that the system must be connected (i.e., all nodes connected). This condition is guaranteed by (2) and (3). Thus, satisfying (2), (3), and (8) ensures that any viable solution, as well as the optimal solution, will be radial (Lavorato et al. 2012).

3 Clonal Selection Algorithm (CLONALG)

The CLONALG was originally proposed in de Castro and Von Zuben (2000), inspired by the biological principle of the clonal selection of B lymphocytes in an organism (biological immune systems). Applying the CLONALG to optimization problems can be described as presented in Pseudocode 1 (de Castro and Von Zuben 2000).

The antibodies correspond to candidate solutions. Each antibody generates a total number of clones (Nc). The number of clones Nc used in the cloning process (line 5 of Pseudocode) for each antibody is given by (13) (de Castro and Timmis 2002). In this expression, β is a cloning factor between 0 and 1, N is the total number of antibodies of population P, and round($\cdot$) is the rounding operator to the nearest integer.

$$N_c^i = \text{round}\left(\frac{\beta N}{i}\right) \quad (13)$$

	Pseudocode A: CLONALG Algorithm
1	Generate a population (P) with N antibodies interpreted as candidate solutions to the problem under analysis;
2	**while** (stopping criterion is not satisfied) **do**
3	Evaluate the affinity (objective function) for each antibody of the population P ;
4	Select the n best antibodies of population P (selection process), obtaining the set $P_{\{n\}}$;
5	Reproduce the n best antibodies selected (cloning process), generating a population (C) with Nc clones. The quantity of clones of each antibody is directly proportional to its affinity;
6	Submit the population of clones (C) to a hypermutation process in which the mutation rate is inversely proportional to the antibody's affinity. A population (C^*) of antibodies is generated;
7	Evaluate the affinity of each antibody that belongs to the set (C^*), and re-select the n best antibodies creating the set $C^*_{\{n\}}$, and add them to population P;
8	Replace d low affinity antibodies in population P with new antibodies ($P_{\{d\}}$) (metadynamic process). The antibodies with low affinity have a greater probability of being replaced;
9	**end while**

The mutation rate (α) of each clone is defined by (14) (de Castro and Timmis 2002). In this expression, ρ is a damping control parameter for the exponential function, and f^* is the normalized value of affinity f, which is calculated according to (15) for minimization problems and according to (16) for maximization problems (de Castro and Timmis 2002).

$$\alpha = \exp(-\rho f^*) \quad (14)$$
$$f^* = \frac{f}{f_{\max}} \quad (15)$$
$$f^* = \frac{f_{\min}}{f} \quad (16)$$

where $f_{\max}$ is the largest affinity and $f_{\min}$ is the lowest affinity.

Each clone suffers a mutation process as proposed in Franca et al. (2005):

$$m = \text{round}(\alpha^* N(0, 1)) \quad (17)$$

where m is the number of mutations that each antibody will suffer, α is the mutation rate, and $N(0, 1)$ is a Gaussian

random variable with a mean of 0 and standard deviation of 1.

The stopping criterion usually employed in the CLON-ALG is a maximum number of iterations.

4 Specialized Genetic Algorithm of Chu–Beasley (SGACB)

The SGACB was originally proposed in Silva et al. (2005) and corresponds to a specialized version of the genetic algorithm of Chu–Beasley presented in Chu and Beasley (1997). The SGACB can be described in Pseudocode 2 (Silva et al. 2005):

The stopping criterion commonly employed in the SGACB is a maximum number of iterations

Pseudocode B: SGACH Algorithm	
1	Generate a population (P) with N individuals;
2	**while** (stopping criterion is not satisfied) **do**
3	Evaluate the fitness (objective function) and the unfitness for each individual in population P;
4	Perform the tournament selection process, generating a pair of individuals (parent individuals) in the population (Ph_1 and Ph_2);
5	Perform the recombination process, generating two child individuals (Ch_1 and Ch_2) from the parents;
6	Evaluate the fitness function of each child (Ch_1 and Ch_2). Eliminate the one having the largest fitness value from the generated pair and keep the one having the lowest fitness value;
7	Perform the mutation process on the child that was selected in the previous step. In this process the child will be mutated, generating a matured child (Ch*);
8	Submit the matured child (Ch*) to an improvement process using a local heuristic search. In this step, the matured child passes through a local search in order to improve the fitness value;
9	Evaluate if the improved child can go back into population P, substituting some individuals in the current population. In the first place, check if this improved child is already in the current population, if yes, the improved child is discarded. Otherwise, the fitness and unfitness values are used to verify the possibility of substitution, depending on the problem.
10	**end while**

5 Codification of the DSR Problem with Variable Demands

In this paper is used the coding of DSR problem proposed in Carreño et al. (2008), wherein the EDS is represented as a tree (graph theory) constituted by an array of arcs (branches). The encoding vector has dimension n_l (number of branches) and represents the whole electrical system storing the branches of the system. In this coding scheme, the first $(n_b - 1)$ elements of the vector indicate the branches of the radial topology (set $N1$), and the branches between position n_b and n_l (set $N2$), indicate the connection elements (branches off the radial configuration), as illustrated in Fig. 1.

For example, a topology of the test system with 14 nodes is shown in Fig. 2. This topology can be represented as (18), where all elements between 1 and $(n_b - 1)$ belong to the network topology (continuous line), and the others, from n_b until the end of the vector, are connection elements that are disconnected in this configuration (dotted line). This representation is not fixed, given that the same topology can be represented as (19), which can help to diversify the search depending on the kind of operators that are used.

$$[C_1 \ C_5 \ C_{10} \ C_2 \ C_7 \ C_{11} \ C_3 \ C_6 \ C_{12} \ C_4 \ C_8 \ C_{13} \ C_9 \ (C_{14} \ C_{15} \ C_{16})] \tag{18}$$
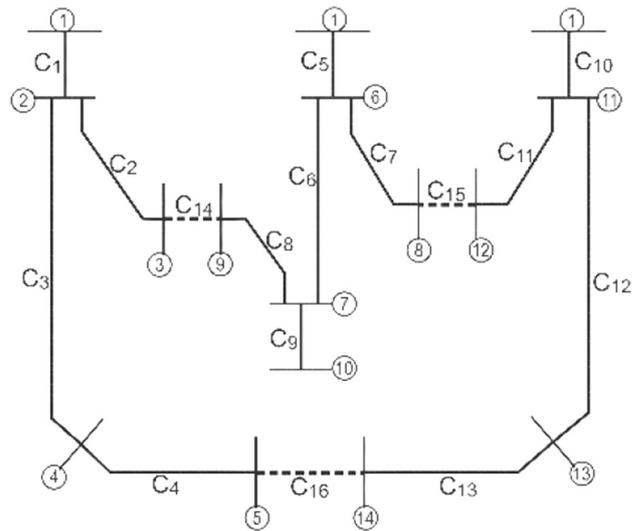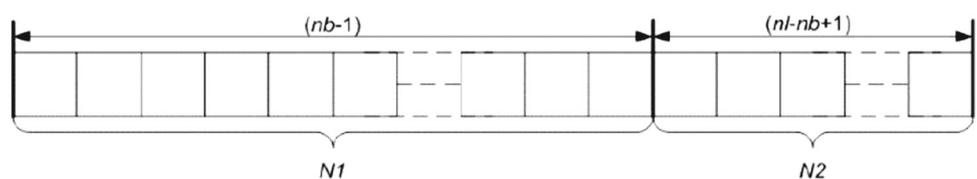

Fig. 2 Example of codification to the 14-node system

Fig. 1 Codification vector for the 14-node system

$$[C_1 \ C_2 \ C_3 \ C_4 \ C_5 \ C_6 \ C_7 \ C_8 \ C_9 \ C_{10} \ C_{11} \ C_{12}$$
$$C_{13} \ (C_{14} \ C_{15} \ C_{16})] \tag{19}$$

This codification ensures that the initial population only includes radial topologies. Therefore, it is possible to develop search operators that preserve the radial topology, avoiding this type of infeasibility.

6 Proposed Approach to the DSR Problem

In this section, two new methods are proposed to solve the DSR problem with variable demands. These methods use the CLONALG (described in Sect. 6.1) and the SGACB (described in Sect. 6.2).

6.1 Application of the CLONALG

In Fig. 3 is illustrated the flowchart of the application of the CLONALG in the RSD problem.

The following sections describe the operators and process of the CLONALG.

6.1.1 Heuristic to Generate the Initial Population

To generate the initial population (P) for the CLONALG was used a heuristic inspired by Prim heuristic presented by Carreño et al. (2008). This heuristic considers the codification of the problem presented in the previous section and described in Pseudocode 3.

Pseudocode C: Heuristic to generate initial population
1   D is the set of nodes present in the current configuration;
2   $D = \phi$, $N1 = \phi$, $N2 = \phi$;
3   Add the node to the substation (root node) to D;
4   Find all branches with a vertex in D;
5   Select a branch, identified in the previous step to enter in the topology (randomly);
6   If the selected branch generates a loop in the topology, include it in $N2$, otherwise include the opposite vertex in D and the branch in $N1$;
7   Back to step 3 until all branches are selected.

In the end of the process, the vector $[N1, N2]$ represents a topology of the EDS. This strategy always generates topologically feasible solutions (radial), contributing to the efficiency and diversity of the algorithms proposed in this paper.

For more details of the heuristic to generate the initial population, consult the Ref. Carreño et al. (2008), where it is detailed in step by step.
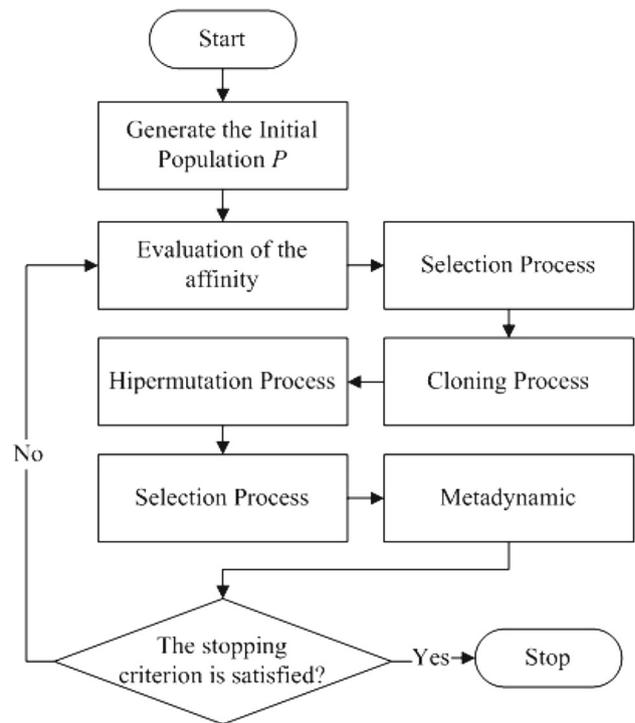

Fig. 3 Flowchart of application of the CLONALG

6.1.2 Evaluation of the Affinity

In this step, the affinity of each antibody (candidate solution) in population P is evaluated. This value represents the cost of the energy losses associated with a topology in the operation period for all demand levels under analysis. In order to obtain the cost of losses, a specialized sweep load flow for radial EDSs (Shirmohammadi et al. 1988) is run. To analyze several demand levels, the power flow is run for each demand, and then expression (20) is used to obtain the affinity value of the associated antibody. In this expression, Nd is the number of demand levels to be analyzed, K_i is the cost of energy losses for demand level i, and T_i and P_i are the duration and the value, respectively, of the active power losses associated with demand level i.

$$f = \sum_{i=1}^{Nd} [K_i * T_i * P_i] \tag{20}$$

Additionally, for the CLONALG, a value of the penalty for each antibody in the population is calculated by measuring its infeasibility degree for all the demand levels to be analyzed. This value is given by the sum of the absolute values of the deviations of nodal voltages in terms of the specified ranges plus the sum of the deviations of the branch currents in terms of the thermal limits of each branch. If this value is zero, then the associated antibody is not penalized and can

be implemented from a technical point of view at all demand levels.

6.1.3 Selection Process

In the selection process of the CLONALG, the n best antibodies (memory antibodies) are selected to carry out the cloning and hypermutation processes, as well as the re-selection mechanism to insert the n best matured antibodies into the population (P). In order to select the n best antibodies, the algorithm takes into account the affinity values ordered from the one with the least power losses (the one of highest quality) to the one with the most power losses. The selected subpopulation of antibodies is called $P_{\{n\}}$.

6.1.4 Cloning Process

In the cloning process, a subpopulation of clones (C) is generated. The subpopulation consists of Nc clones obtained through the antibodies of subpopulation $P_{\{n\}}$. The number of clones that each antibody can generate is calculated by using (13).

6.1.5 Hypermutation Process

After obtaining a subpopulation of clones (C), the hypermutation process is carried out to generate a new subpopulation of matured clones (C^*). Initially it is necessary to calculate the number of mutations of an antibody i in the iterative process of the CLONALG using (17). Equation 14) defines the mutation rate (α), and (16) establishes the normalized affinity (f^*). Following these calculations, a random mutation is performed as described as following Pseudocode.

	Pseudocode D: Hypermutation process
1	Read C and obtain Nc (number of clones of population C);
2	**for** i=1 until Nc **do**
3	Calculate the number of mutations (m) for antibody i;
4	**for** j=1 until m **do**
5	Select randomly a circuit $l \in N2$ of antibody i. This position l represents a disconnected circuit;
6	Connect/close the selected circuit and identify the loop formed. Choose a circuit in the loop formed and replace by the circuit l;
7	**end for**
8	Store the matured antibody i in C^*;
9	**end for**
10	**return** C^*;

Figure 4 presents an illustrative example of the hypermutation process. In this example it was used the antibody shown in (19). In this process, it was initially randomly cho-
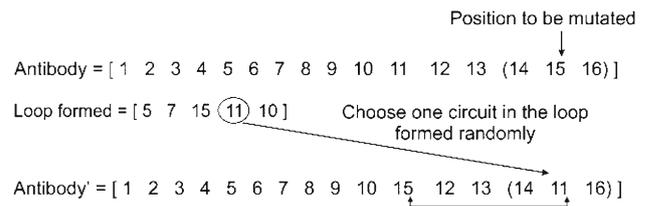

Fig. 4 Hypermutation example for the 14-node system

sen a circuit belonging to set $N2$, i. e., a disconnected circuit. After this process, this circuit is connected and a loop is formed in the system. This loop should be identified and stored. In the sequence, it is randomly chosen another circuit belonging to the mentioned loop formed to replace the initial circuit. Finally the circuits are exchanged (the circuit 15 is replaced by the circuit 11), generating a matured antibody. This hypermutation strategy always generates topologically feasible antibodies.

6.1.6 Metadynamic Process

In the metadynamic process of the CLONALG, the worst d antibodies of the population (P) are replaced by d new antibodies, which are randomly generated using the heuristic described in Pseudocode 3, presented in Sect. 6.1.1.

6.1.7 Stopping Criterion

The CLONALG stops if the affinity value of the best antibody in the population does not change for a specified number of iterations (ger), and the average affinity value of the antibodies in the population does not change by more than a specified percentage over a given number of iterations. If the above two conditions hold, then the algorithm ends, indicating that a final solution has been obtained. If not, the iteration counter is increased by 1. If the maximum number of iterations has been reached, then stop. If not, return to Step 2 of the algorithm.

6.2 Application of the SGACB

In Fig. 5 is illustrated the flowchart of the application of the SGACB in the RSD problem.

The following sections describe the SGACB. In order to conduct a comparison between the CLONALG and the SGACB, both were implemented using the same strategies for codification, generation of the initial population, evaluation of the affinity (objective function), and mutation.

6.2.1 Heuristic to Generate the Initial Population

The heuristic to generate the initial population of the SGACB is the same as the one used to generate the initial population for the CLONALG, as presented in Sect. 6.1.1.
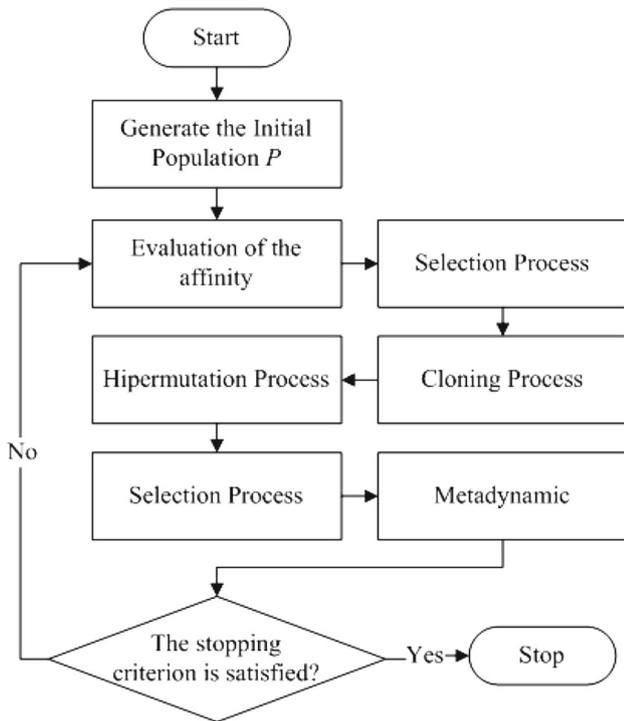
Fig. 5 Flowchart of application of the SGACB

Pseudocode E: Recombination process	
1	Read the individual parents;
2	Choose randomly a cross point (*nr*) in the set of *N1* of the first parent;
3	Copy the first *nr* elements of the first parent to the child individual in construction;
4	In the second parent, eliminating all the elements that are already present in the child individual;
5	For each branch still present in the second parent: If the branch does not form a loop, include in the set of N1, otherwise, include in the set of N2.

6.2.2 Evaluation of Fitness and Unfitness

To evaluate the fitness value for the SGACB, the same operator as the one described in Sect. 6.1.2 of the CLONALG is used. In the SGACB, unfeasibility is represented by the unfitness (UF) coefficient and used in the substitution step of the SGACB (Sect. 6.2.7).

6.2.3 Selection Operator

The selection operator used in the SGACB is based on the tournament selection process. This mechanism uses pairs of individuals in population P to select two individuals (Ph_1 and Ph_2) to be processed by the algorithm.

6.2.4 Recombination Process

After the selection process, the recombination step takes place, generating two child individuals from the parent individuals. This process is presented in Carreño et al. (2008) and described in the following steps.

This recombination strategy generates only child individuals that are topologically feasible; that is, they remain associated with radial topologies.

6.2.5 Mutation Process

The mutation process used in the SGACB is the same as the one described in Sect. 6.1.5 of the CLONALG.

6.2.6 Improvement Process

The improvement process used in the SGACB is an intensified search (local improvement), which is run to improve the fitness values of the individuals. This process uses the local search heuristic proposed in Carreño et al. (2008) and detailed in Souza et al. (2015b).

6.2.7 Substitution Criterion

The substitution criterion of the SGACB evaluates whether the improved child can substitute any individual in the current population. The criterion to make this substitution is as follows.

First, it is checked whether this improved child is already in the current population P. If it is, then the improved child is eliminated. If not:

(a) If the improved child corresponds to a solution that is unfeasible (i.e., UF is not zero), then it is only possible to substitute an individual in the population that is also unfeasible. The substitution takes place if the degree of unfeasibility (UF) of the mutated child is smaller than the UF value of the unfeasible individual in the current population that has the largest UF. If the UF of the mutated child is larger than the largest UF of the unfeasible individuals in the current population, then the mutated child is eliminated;

(b) If the mutated child is feasible (i.e., UF is zero) and there are unfeasible individuals in the current population,

Fig. 6 Typical active power demand load curves of **a** residential, **b** commercial and **c** industrial
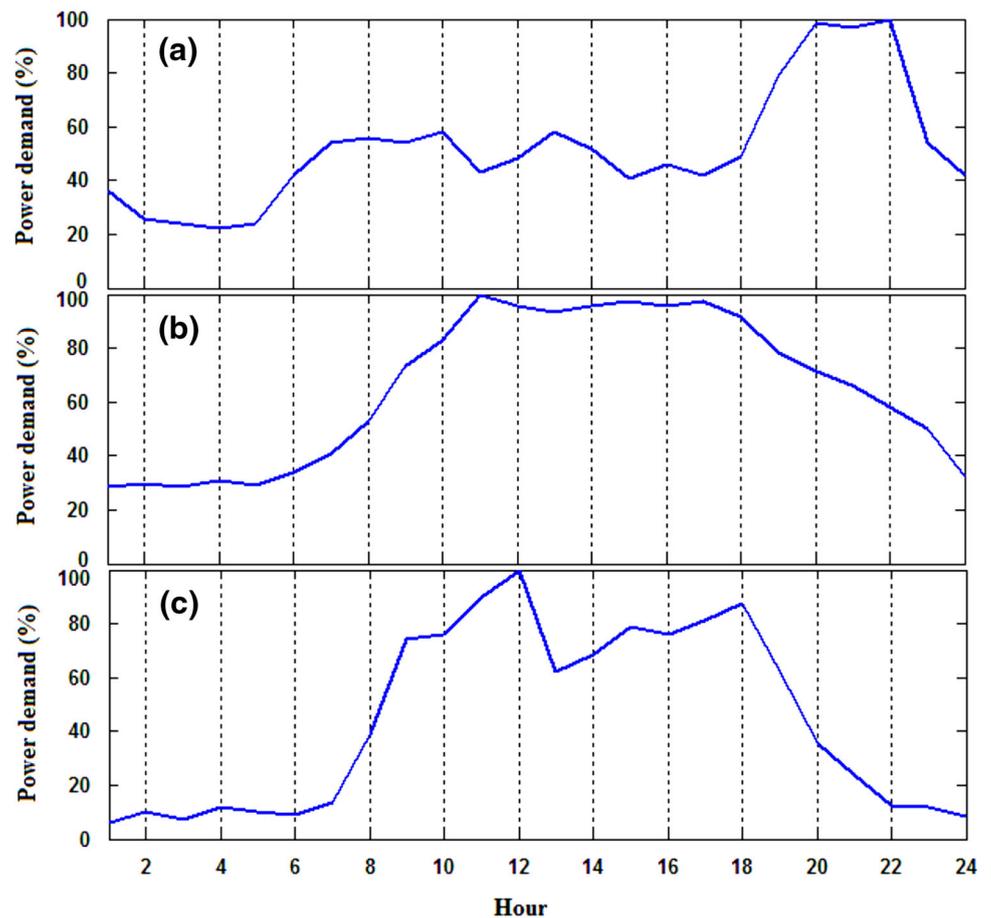
then the mutated child replaces the one with the largest UF, that is, the most unfeasible individual in the current population. If no unfeasible individuals exist in the current population, then the mutated child replaces the worst of the individuals in the population, provided that its fitness value is smaller. If not, the mutated child is eliminated.

6.2.8 Stopping Criterion

The stopping criterion used in the SGACB is the same as the one used in the CLONALG (Sect. 6.1.7).

7 Results

This section presents the results obtained from the application of the CLONALG and the SGACB to the test systems with 33 (Baran and Wu 1989), 84 (Chiou et al. 2005), and 136 (Zhang et al. 2007) nodes, and to the real system with 417 (Bernal-Agustin 1998) nodes. The CLONALG and the SGACB were implemented in C++ 6.0® (Borland C++). All tests were carried out using a PC with an Intel Core i7 3.1 GHz processor.

7.1 Configuration of the Used Demand Levels

In the all tests is considered an operation period of 24 h, and for each hourly period, a demand level was specified for each node in the test systems. The demand levels were organized into three sets: (a) residential, (b) commercial, and (c) industrial. For each demand level, a typical load diagram was specified, as illustrated in Fig. 6a–c.

In order to obtain one specific demand level, the load factor of the final consumer at each node was multiplied by the active and reactive powers associated with the typical load curves. The selection of the type of consumer at each node of the EDS was done in a random way, assuming that 60 % of the consumers were residential, 25 % were commercial, and 15 % were industrial.

The types of consumer defined for each bar of the EDS can be found in Possagnolo (2015), where is detailed the data of the EDS used in this paper.

Table 1 presents the load factors associated with each type of consumer over the 24 h of the operation period and the cost of energy losses. Figure 7 shows how the cost of energy losses varies across the 24-h period. In each hourly period, these costs are multiplied by the active losses calculated by the power flow routine in order to obtain the

Table 1 Load factors of the consumers and cost of energy losses

Demand level	Consumer kind			Cost (US$/kWh)
	Residential	Commercial	Industrial	
1	0.3600	0.2838	0.0625	0.0650
2	0.2600	0.2973	0.1000	0.0650
3	0.2400	0.2838	0.0750	0.0650
4	0.2200	0.3108	0.1188	0.0650
5	0.2400	0.2938	0.1000	0.0650
6	0.4200	0.3378	0.0875	0.0650
7	0.5400	0.4054	0.1375	0.1100
8	0.5600	0.5270	0.3875	0.1100
9	0.5400	0.7297	0.7438	0.1100
10	0.5800	0.8311	0.7625	0.1100
11	0.4300	1.0000	0.9000	0.1100
12	0.4800	0.9595	1.0000	0.1100
13	0.5800	0.9324	0.6188	0.1100
14	0.5200	0.9595	0.6875	0.1100
15	0.4100	0.9730	0.7875	0.1300
16	0.4600	0.9595	0.7625	0.1300
17	0.4200	0.9730	0.8125	0.1300
18	0.4900	0.9189	0.8750	0.1300
19	0.7900	0.7838	0.6188	0.1500
20	0.9840	0.7162	0.3563	0.1500
21	0.9700	0.6622	0.2375	0.1500
22	1.0000	0.5811	0.1250	0.0650
23	0.5400	0.5000	0.1188	0.0650
24	0.4200	0.3229	0.0832	0.0650
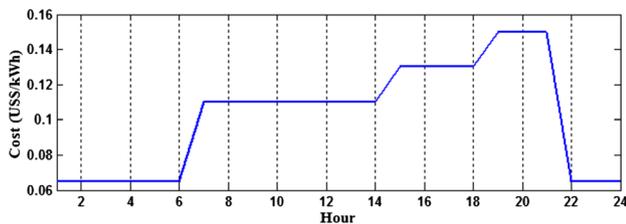
Fig. 7 Cost of the energy losses (US$/kWh)

value of the objective function associated with each candidate solution.

It should be noted that in this paper was used hypothetical curves to the power load and cost of losses. This curves, were obtained in the literature (Possagnolo 2015). In a real situation, the distribution company should be able to estimate all these data.

7.2 Parameters Used in the Algorithms

The results for all of the systems were obtained using the parameters shown in Table 2.

In Table 2, N is the number of antibodies/individuals in the population, β is the cloning factor, ger is the number of generations, n is the number of memory antibodies, d is the number of antibodies that will be replaced in the metadynamic process, ρ is the damping control parameter for the exponential function, and iter is the number of iterations.

These parameters were established after running a number of trial tests. Several values were tested for each parameter, and the value that provided the best performance was selected. These tests were carried out until the most suitable set of parameters for each distribution system was determined.

It must be noted that the parameters used in the CLONALG to solve the test systems differed only in the number of generations (ger) and the number of memory antibodies (n), which demonstrates the robustness of this method. In the same way, the parameters used in the SGACB differed only in the number of iterations (iter) and the number of individuals (N) in the population.

7.3 33-, 84-, 136-, and 417-Node Systems

The well-known 33-, 84-, and 136-node test systems and the 417-node real system had nominal voltages of 12.66, 11.40, 13.80, and 11.00 kV, respectively.

Table 3 shows the initial and final topologies (i.e., before and after reconfiguration) found using the CLONALG and the SGACB approaches, along with the cost of energy losses for each EDS tested. In all tests, the initial and final topologies represent a unique topology that should be used in the 24 h of operation of the EDS. For all of the tests, both algorithms led to a reduction of between 10 and 31 % in the cost of energy losses.

In order to evaluate the performance of the proposed algorithms, the CLONALG and the SGACB were each executed 30 times. In all executions, the algorithms found the best solution for all systems, but with a different number of iterations and computational time.

Tables 4 and 5 present a statistical analysis performed with the number of iterations and the computational time required for the 30 executions to find the solution for all of the systems.

Table 4 includes data regarding iTmax, iTmin, iTean, and dev, representing the larger, the lower, the mean, and the standard deviation of the execution time for the number of iterations needed to find the best solution, respectively.

Table 5 presents four types of data. Tmax and Tmin are the maximum and the minimum time, respectively, required to find the solution, while Tmean is the mean time taken by the 30 executions, and dev is the standard deviation of that number, respectively.

A comparative analysis of the data in Tables 4 and 5 reveals the differences between the CLONALG and SGACB methods. The CLONALG was solved with fewer iterations and

Table 2 Parameters used in the algorithms

Parameters		N	β	ger	n	d	ρ
CLONALG	33-, 84-, 136-node systems	50	0.5	50	10	1	4
	417-node system	50	0.5	80	30	1	4
Parameters		N	iter				
SGACB	33-, 84-, 136-node systems	50	100				
	417-node system	80	200				

Table 3 Results for the 33-, 84-, 136-, and 417-node systems

System	Before reconfiguration		After reconfiguration		Reduction (%)
	Topology	Cost of energy losses (US$)	Topology	Cost of energy losses (US$)	
CLONALG and SGACB					
33	33-34-35-36-37	187.86	7-9-14-28-32	128.81	31.43
84	84-85-86-87-88-89-90-91-92-93-94-95-96	456.41	7-34-39-63-72-83-84-86-88-89-90-92-95	410.53	10.05
136	136-137-138-139-140-141-142-143-144-145-146-147-148-149-150-151-152-153-154-155-156	288.50	7-38-51-54-84-90-96-106-118-126-135-137-138-141-144-145-147-148-150-151-155	256.89	10.95
417	1-5-15-16-26-31-53-54-55-75-82-94-96-97-106-107-119-136-138-154-155-156-168-169-177-179-194-195-201-207-211-214-219-241-256-258-282-297-302-314-321-354-359-362-364-385-388-395-396-404-407-423-424-426-431-436-445-446-449	637.88	1-2-13-15-16-26-31-40-41-50-59-73-82-94-96-97-111-115-136-146-150-155-156-158-163-168-169-178-179-190-191-194-195-209-230-254-256-267-270-294-310-321-354-362-385-389-392-395-403-404-423-424-426-436-437-439-446-449-466	529.66	16.96

Table 4 Number of iterations needed to find the solutions

Algorithm	CLONALG				SGACB			
System	iTmin	iTmax	iTmean	dev	iTmin	iTmax	iTmean	dev
33	1	3	1.43	0.72	1	10	4.83	2.42
84	2	7	3.70	1.59	8	21	17.20	3.94
136	11	24	16.17	4.55	29	69	49.30	11.15
417	25	61	47.13	12.74	74	158	124.47	26.28

Table 5 Computational time needed to find the solutions

Algorithm	CLONALG				SGACB			
System	Tmin (s)	Tmax (s)	Tmean (s)	dev	Tmin (s)	Tmax (s)	Tmean (s)	dev
33	0.245	0.326	0.295	0.022	0.297	0.412	0.372	0.034
84	3.568	4.315	3.995	0.180	3.957	5.236	4.824	0.330
136	18.235	22.153	20.401	0.984	21.964	26.458	24.796	1.005
417	188.124	192.235	190.301	1.090	239.248	262.345	253.289	6.427

Table 6 Comparison with the literature

System	Topology	Cost of energy losses (US#)	Reference	Kind of analysis
33	7-9-14-28-32	128.81	Possagnolo (2015)	Variable demand
	7-9-14-32-37	134.30	Souza et al. (2015b)	Fixed demand
84	7-34-39-63-72-83-84-86 -88-89-90-92-95	410.53	Possagnolo (2015)	Variable demand
	7-13-34-39-42-55-62-72- 83-86-89-90-92	417.29	Souza et al. (2015b); Lavorato et al. (2012)	Fixed demand
136	7-38-51-54-84-90-96-106- 118-126-135-137-138- 141-144-145-147-148- 150-151-155	256.89	Possagnolo (2015)	Variable demand
	7-35-51-90-96-106-118- 126-135-137-138-141- 142-144-145-146-147- 148-150-151-155	272.96	Carpaneto and Chicco (2008); Lavorato et al. (2012)	Fixed demand
417	1-2-13-15-16-26-31-40-41- 50-59-73-82-94-96-97- 111-115-136-146-150- 155-156-158-163-168- 169-178-179-190-191- 194-195-209-230-254- 256-267-270-294-310- 321-354-362-385-389- 392-395-403-404-423- 424-426-436-437-439- 446-449-466	529.66	Possagnolo (2015)	Variable demand
	5-13-15-16-21-26-31-54- 57-59-60-73-86-87-94- 96-97-111-115-136-142- 149-150-155-156-158- 163-168-169-178-179- 191-195-199-214-221- 254-256-266-282-317- 322-325-358-362-369- 392-395-403-404-416- 423-426-431-436-437- 446-449-466	530.29	Souza et al. (2015b)	Fixed demand

less computational time than the SGACB. However, both methods found the same topology and therefore obtained the same final value for the objective function.

7.4 Comparative Analysis with the Literature

Table 6 presents the best solutions available in the literature. Comparisons were performed with the results considering fixed and variable demands. To obtain the value of the cost of energy for the fixed demands, the topologies presented in these papers were used, along with the affinity/fitness function to obtain the value of the cost.

For the 33-, 84-, 136- and 417-node systems, both algorithms found the best solutions reported in the literature, as presented in Table 6. From these comparisons, it can be concluded that the CLONALG and SGACB are efficient methods because they arrived at the same solutions as the

best solutions found in the specialized literature. It should be noted that there are few studies in the literature considering variable demands, as presented in this paper, which makes it difficult to conduct a more extensive comparison of our results.

7.5 Comments about the CLONALG and SGACB Methods

After carrying out the tests and obtaining the results reported above were identified the strengths and weaknesses of the proposed algorithms for solving the DSR problem with variable demands. These are detailed below.

Strengths:

- The proposed algorithms showed excellent performance, finding the same solutions as the best solutions reported in the specialized literature;

- Both algorithms require low processing time and a reduced number of iterations;
- The CLONALG and the SGACB include efficient strategies that enable the maintenance of population diversity and keep the incumbent (i.e., the best solution found in the process) within the population;
- The algorithms are robust and converged to the best result in all tests that were performed.

Weakness:

- The CLONALG requires the setting of several parameters in advance.

8 Conclusions

This paper presented two new approaches to solve the DSR problem with variable demands using intelligent algorithms. The algorithms were applied to identify topological configurations that would minimize the cost of energy losses in radial distribution systems. When carrying out the tests, both algorithms found the best solutions available in the literature. The CLONALG achieved the solution using fewer iterations and less computational time than the SGACB.

The CLONALG method is more efficient due to the strategies used to intensify the search in the solution space of the problem. Nevertheless, the SGACB also found the solution, despite the greater number of iterations, and so can be used as an efficient alternative. The CLONALG and the SGACB were stable and reliable in solving the DSR problem with variable demands; for all performed runs, they found the best solutions for all of the test systems. The results obtained for all systems were compared to the solutions available in the literature, thereby proving the efficiency of the proposed methods.

This work contributes two new proposed methods to assist professionals in carrying out the planning and operation of EDSs. It was concluded that the proposed CLONALG and SGACB methods for solving the DSR problem with variable demands performed well, with efficiency, robustness, and low computational time.

References

Abdelaziz, A. Y., Mohammed, F. M., Mekhamer, S. F., & Badr, M. A. I. (2009). Distribution system reconfiguration using a modified particle swarm optimization algorithm. *Electric Power Systems Research*, 79, 1521–1530.

Abdelaziz, A. Y., Mohammed, F. M., Mekhamer, S. F., & Badr, M. A. I. (2010). Distribution system reconfiguration using a modified tabu search algorithm. *Electric Power Systems Research*, 80, 943–953.

Baran, M. E., & Wu, F. F. (1989). Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transactions on Power Delivery*, 4(2), 1401–1407.

Bernal-Agustin, J. L. (1998) Application of genetic algorithms to the optimal design of power distribution systems. 346 f. Tesis, University of Zaragoza, Zaragoza.

Borland 6.0 Version, C++ Builder.

Bueno, E. A., Lyra, C., & Cavellucci, C. (2004). Distribution network reconfiguration for loss reduction with variable demands. In *IEEE Latin America transmission and distribution conference* (pp. 384–389). Colombia: Medellin.

Carpaneto, E., & Chicco, G. (2008). Distribution system minimum loss reconfiguration in the hyper-cube ant colony optimization framework. *Electric Power Systems Research*, 78, 2037–2045.

Carreño, E. M., Romero, R., & Feltrin, A. P. (2008). An efficient-codification to solve distribution network reconfiguration for lossreduction problem. *IEEE Transactions on Power Systems*, 23(4), 1542–1551.

Chang, H. C., & Kuo, C. C. (1994). Network reconfiguration in distribution systems using simulated annealing. *Electric Power Systems Research*, 29(3), 227–238.

Chiou, J. P., Chang, C. F., & Su, C. T. (2005). Variable scaling hybrid differential evolution for solving network reconfiguration of distribution systems. *IEEE Transactions on Power Systems*, 20(2), 668–674.

Chu, P., & Beasley, J. E. (1997). A genetic algorithm for the generalized assignment problem. *Computers and Operations Research*, 24(1), 17–23.

Civanlar, S., Grainger, J. J., & Lee, S. S. H. (1988). Distribution feeder reconfiguration for loss reduction. *IEEE Transactions on Power Delivery*, 3(3), 1217–1223.

de Castro, L. N., & Timmis, J. (2002). *Artificial immune systems: A new computational intelligence approach* (1st ed.). Berlin: Springer.

de Castro, L. N., & Von Zuben, F. J. (2000) The clonal selection algorithm with engineering applications. In *Workshop on artificial immune systems and their applications* (pp. 36–37).

de Franca, F. O., Von Zuben, F. J., & de Castro, L. N. (2005). An artificial immune network for multimodal function optimization on dynamic environments. In *GECCO*. Washington, DC, USA (pp. 289–296).

Franco, J. F., Lavorato, M., Rider, M. J., & Romero, R. (2012). An efficient implementation of tabu search in feeder reconfiguration of distribution systems. In *IEEE PES general meeting*, San Diego, California, USA (pp. 1–8).

Guimarães, M. A. N., Castro, C. A., & Romero, R. (2010). Distribution system operation optimisation through reconfiguration and capacitor allocation by a dedicated genetic algorithm. *IET Generation, Transmission & Distribution*, 4(11), 1213–1222.

Inoue, T., Takano, K., Watanabe, T., & Kawahara, J. (2014). Distribution loss minimization with guaranteed error bound. *IEEE Transactions on Smart Grid*, 5(1), 102–111.

Kavousi-Fard, A., & Niknam, T. (2014). Optimal distribution feeder reconfiguration for reliability improvement considering uncertainty. *IEEE Transactions on Power Delivery*, 29(3), 1344–1353.

Kim, H., Ko, Y., & Jung, K. H. (1993). Artificial neural network based feeder reconfiguration for loss reduction in distribution systems. *IEEE Transactions on Power Delivery*, 8(3), 1356–1366.

Lavorato, M., Franco, J. F., Rider, M. J., & Romero, R. (2012). Imposing radiality constraints in distribution system optimization problems. *IEEE Transaction on Power Systems*, 27(1), 172–180.

Merlin, A., & Back, H. (1975) Search for a minimal-loss operating spinning tree configuration in an urban power distribution system.

In *Power System computation conference*, Cambridge, UK (pp. 1–18).

Nara, K., Shiose, A., Kitagawa, M., & Ishihara, T. (1992). Implementation of genetic algorithm for distribution systems loss minimum reconfiguration. *IEEE Transactions on Power Systems*, 7(3), 1044–1051.

Possagnolo, L. H. F. M. (2015) Distribution systems reconfiguration operating in several demand levels through of the variable neighborhood search (Master's Thesis), Campus of Ilha Solteira, Unesp, Univ Estadual Paulista **(in Portuguese)**.

Queiroz, L. M. O., & Lyra, C. (2009). Adaptive hybrid genetic algorithm for techinical loss reduction in distribution networks under variable demands. *IEEE Transactions on Power Systems*, 24(1), 445–453.

Salazar, H., Gallego, R., & Romero, R. (2006). Artificial neural networks and clustering techniques applied in the reconfiguration of distribution systems. *IEEE Transactions on Power Delivery*, 21(3), 1735–1742.

Shirmohammadi, D., Hong, H. W., Semlyen, A., & Luo, G. X. (1988). A compensation based power flow method for weakly meshed distribution and transmission networks. *IEEE Transactions on Power Systems*, 3(2), 753–762.

Silva, I. J., Rider, M. J., Romero, R., Garcia, A. V., & Murari, C. A. (2005). Transmission network expansion planning with security constraints. *IEE Proceedings - Generation, Transmission and Distribution*, 152(6), 828–836.

Souza, S. S. F., Romero, R., Pereira, J., & Saraiva, J. T. (2015a). Specialized genetic algorithm of Chu-Beasley applied to the distribution system reconfiguration problem considering severaldemand scenarios. In *IEEE powertech*, Eindhoven (pp. 1–6).

Souza, S. S. F., Romero, R., & Franco, J. F. (2015b). Artificial immune networks Copt-aiNet and Opt-aiNet applied to the reconfiguration problem of radial electrical distribution systems. *Electric Power Systems Research*, 119, 304–312.

Zhang, D., Fu, Z., & Zhang, L. (2007). An improved TS algorithm for loss-minimum reconfiguration in large-scale distribution systems. *Electric Power Systems Research*, 77(5–6), 685–694.