Paper accepted for presentation at 2009 IEEE Bucharest Power Tech Conference, June 28th - July 2nd, Bucharest, Romania

Energy Restoration in Distribution Systems using Multi-Objective Evolutionary Algorithm and an Efficient Data Structure

M. R. Mansour, A. C. Santos, J. B. London Jr., A. C. B. Delbem, N. G. Bretas

***Abstract*—This paper proposes a new strategy for solving the service restoration problem in large-scale Distribution Systems (DS). Due to the presence of various conflicting objective functions and constraints, the service restoration task is a multi-objective, multi-constraint optimization problem. As a consequence, finding feasible solutions is a hard task. The proposed strategy uses a new tree encoding, called Node-depth Encoding (NDE), and a modified version of the Non-Dominated Sorting Genetic Algorithm-II (NSGA-II). Using NDE and its operators the proposed strategy generates only radial configurations without disconnected areas reducing the running time necessary to find feasible solutions. On the other hand, the use of the modified version of the NSGA-II enables an efficient exploration of the search space. The efficiency of the proposed strategy is shown using a Brazilian DS, with 3,860 buses, 635 switches, 3 substations, 23 feeders, 2 transformers of 50MVA and 1 transformer of 25MVA.**

***Index Terms*—Energy Restoration, Large-Scale Distribution System, Data Structure, Node-depth Encoding, Multi-Objective Evolutionary Algorithms, NSGA-II.**

I. Introduction

AFTER a faulted zone has been identified and isolated, an energy restoration plan (ERP) for distribution systems (DS) plays a key part in improving service reliability and enhancing customers satisfaction. The main objectives of an energy restoration problem are (i) the restoration of energy as much as possible, by transferring loads from the out-of-service areas via network reconfiguration, (ii) using a reduced number of switching operations and (iii) a low running time; while at the same time respecting security constraints. A second importance aspect is the minimization of the number of network losses. In mathematical terms, the service restoration is a multi-objective, multi-constrained, combinatorial, nonlinear and real-time optimization problem.

In order to deal with this complex problem, several Evolutionary Algorithms (EAs) have been developed [1], [2]. The results obtained by such approaches have surpassed those obtained through both Mathematical Programming (MP) and traditional Artificial Intelligence (AI) [3]. However the running time may be very long or even prohibitive in applications of EAs to large-scale networks.

The majority of EAs applied to energy restoration problem are conventional EAs, that is, those that converted a multi-objective optimization problem into a single objective optimization by using weighting factors. The performance obtained by these EAs for energy restoration is drastically affected by the following factors: (i) The adopted data structure: an inadequate representation may reduce the algorithm performance [4]; (ii) The adopted operators, which may produce many configurations that are not feasible (feasible configurations in this context are radial configurations in that all customers are satisfied, after the faulted zone has been isolated [4]); and (iii) The conversion of a multi-objective optimization problem into a single objective optimization is accomplished by using weighting factors [5].

M. R. Mansour, A. C. Santos, J.B.A. London Jr. and N.G. Bretas are with the Department of Electrical Engineering of the EESC - University of Sao Paulo, Brazil (e-mails: {moussa, augustos, jbalj, ngbretas}@sel.eesc.usp.br).

A.C.B. Delbem is with the Computer System Department of the ICMC - University of Sao Paulo, Brazil (e-mail: acbd@icmc.usp.br).

In order to improve the EAs' performance in such problems, reference [4] used a new tree encoding, called Node-depth Encoding (NDE) [6] and a conventional EA with two efficient operators. These operators are able to change the network topology without producing loops or disconnected areas. Simulation results presented in [4] demonstrate that EA is an efficient alternative to deal with energy restoration problem in large-scale DS.

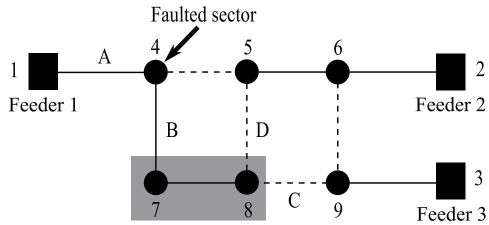
In [7], a multi-objective optimization strategy - Non-Dominated Sorting Genetic Algorithm-II, NSGA-II - is utilized to solve the energy restoration problem. One of the main advantages of this technique is that it deals with many objectives without weight factors. Results presented in [7] have encouraged NSGA-II application in large-scale energy restoration problems.

In this context, a strategy to obtain ERPs for large-scale DS is proposed. The main idea is to utilize NSGA-II with NDE and its operators. Simulation results have demonstrated that the proposed strategy is capable of obtaining efficient solutions when it is applied to large-scale DS in real time.

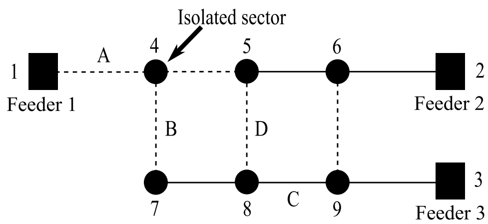
II. Problem Formulation

A. Energy Restorarion

After the location of a fault has been identified and isolated, the out-of-service area must be connected to another feeder by opening and/or closing switches. Fig. 1 shows an example of energy restoration in a DS with three feeders, where black rectangles indicate power sources in a feeder, solid lines are normally closed switches, dashed lines are normally opened switches, and black circles indicate sectors [4]. A sector is a set of connected buses and lines without sectionalizing and tie-switches sectors. Suppose that sector 4 is in fault (Fig. 1(a)). As a consequence, sector 4 must be isolated from the system by opening switches A and B. Sectors 7 and 8 are in an out-of-service area (masked area in Fig. 1(a)). One way to restore energy for these sectors is closing switch C (Fig. 1(b)).



(a) Distribution System in fault.



(b) New Configuration.

Fig. 1. DS Reconfiguration

After a faulted zone has been isolated and the out-ofservice area has been re-connected to the system, the following constraints must be satisfied [4]:

1) A radial network structure;
2) The supply of all possible consumers in the downstream area from the sector in fault;
3) No violation of the current limits in lines and transformers;
4) Voltage drops kept in the adequate range.

Two objectives have been considered for a general energy restoration problem:

1) The reduction of the number of switching operations;
2) The amount of power losses.

B. Mathematical Formulation

Using an Adequate Data Structure (ADS), as well as its operators that generate exclusively feasible configurations and represents the buses in the terminal-substation order (TSO) [8], a general energy restoration problem of DS reconfiguration can be formulated as follows:

$$\begin{aligned} \text{Min.} \quad & E(F) + |\Omega I(F)| \\ \text{s.t.} \quad & \\ & \textit{load flow calculated using an ADS} \\ & F \textit{ is constructed by the ADS operators,} \end{aligned} \tag{1}$$

where:

- F is a graph forest corresponding to a system configuration, where each tree of the forest corresponds to a feeder connected to a substation;
- $E(F)$ is the objective function;
- $I(F)$ represents inequality constraints describing the network operational limits.

Function $E(F)$ contains in general one or more of the following components:

- $\phi(F)$ number of out-of-service loads for a radial network topology (a forest F);
- $\psi(F, F^0)$ number of switching operations required to obtain a given configuration F from the original configuration F^0;
- $\varphi(F)$ system power losses of topology F.

The network operational constraints $I(F)$ in reconfiguration of DS usually include:

- An upper bound of current $\bar{x}_j$ for each line current x_j at line j. The highest ratio $x_j/\bar{x}_j$ is call network loading;
- The maximal current injection $\bar{b}_i$ provided by each substation, where i means substation i. The highest ratio $b_i/\bar{b}_i$ is call substation loading;
- A lower bound for node voltage $\underline{v}$. Let v_i be the node voltage at bus i and vb the system base voltage; the lowest ratio v_i/v_b is call voltage ratio.

The vector of node voltages v is given by $Yv = b$, where Y is the nodal admittance matrix ($Y = AY_xA^T$, where Y_x is the diagonal admittance matrix).

The diagonal matrix Ω is as follows:

$$\mathrm{w}_{11} = \begin{cases} w_x, \text{ if, for at least one } j,\, x_j > \bar{x}_j \\ 0, \text{ otehwise;} \end{cases}$$

$$\mathrm{w}_{22} = \begin{cases} w_s, \text{ if, for at least one } i,\, b_i > \bar{b}_i \\ 0, \text{ otherwise;} \end{cases}$$

$$\mathrm{w}_{33} = \begin{cases} w_v, \text{ if, for at least one } i,\, v_i > \underline{v} \\ 0, \text{ otherwise. ,} \end{cases}$$

and the weights w_x, w_s and w_v are positive values, and $||$ is the L_1-*norm*[1] of a vector.

This paper uses the NDE as an ADS, as it was proposed in [4]. As the NDE operators generate only feasible configurations, it does not require a specific routine to verify or correct unfeasible configuration. The NDE also enables an efficient load flow that fast evaluates each new produced configuration, for large-scale DSs, from the TSO provided by the NDE [9].

The Multi-objective EAs literature shows that aggregation objective function, like $E(F) + |\Omega I(F)|$, restricts largely the search space limiting the quality of the found solutions [5]. Several approaches have been developed to work with several objectives and constraints without such restriction.

In this paper, the aggregation function is decomposed and the problem (See Equation (1)) is reformulated as follows:

$$\begin{aligned} \text{Min.} \quad & E = [e_1(F)|\Omega I'(F)|] \\ \text{s.t.} \quad & \\ & \textit{load flow calculated using an ADS} \\ & F \textit{ is constructed by the ADS operators,} \end{aligned} \tag{2}$$

[1] The L_1-*norm* of a vector z of size n is given by $\sum_{r=1}^{n} |Z_r|$

where E is a vector of two objective functions: $e_1(F)$ is the number of switching operations; and $|\Omega I'(F)|$ is an aggregation function with the remaining constraints and objectives. As the number of manually controlled switches used in DS is bigger than the number of automatic control switches, the time taken by the restoration process depends on the number of switching operations, which should be kept to a minimum possible. As a consequence, we remark that $e_1(F)$ is the most important aspect for the energy restoration problem, thus, it is a separate objective function in the formulation, enabling the searching algorithm to focus on this aspect.

III. Node-Depth Encoding

NDE [6] is a graph encoding based on the concepts of node and node depth in a graph tree and consists basically of a linear list containing the tree nodes and their depths, creating an array of pairs (n_x; d_x), where n_x is a node and d_x its depth. The order in which pairs are placed on the linear list is important. A depth search for a graph spanning tree can produce the proper ordering by inserting a pair (n_x; d_x) in the list when a node nx is visited by the search.

Fig. 2 presents a forest with three spanning trees. The nodes 1, 2 and 3 are the root nodes of trees 1, 2 and 3, respectively. The graph of Fig. 2 can be seen as a DS with 3 feeders, where nodes are sectors, and edges are the sectionalizing switches (edges in solid lines represent normally closed sectionalizing switches and edges in dashed lines represents normally opened tie-switches). The nodes 1, 2 and 3 are, respectively, in substations 1, 2 and 3.

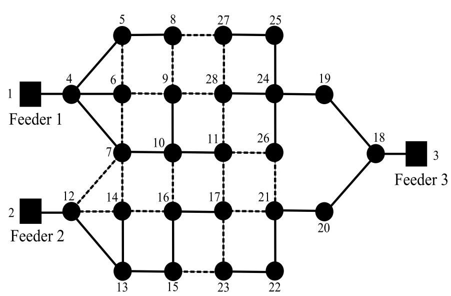


Fig. 2. DS with three feeders. It corresponds to a graph with three trees.

The proposed forest encoding is composed by the union of the enconding of all trees of a forests. Therefore, the forest data structure can be easily implemented using an array of pointers, where each pointer indicates the NDE of a tree. Fig. 3 presents the NDEs for the three trees from Fig. 2.

$$T_1 \begin{bmatrix} \text{depth} \\ \text{node} \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 & 3 & 2 & 2 & 3 & 4 & 4 \\ 1 & 4 & 5 & 8 & 6 & 7 & 10 & 9 & 11 \end{bmatrix}$$

$$T_2 \begin{bmatrix} \text{depth} \\ \text{node} \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 & 3 & 3 & 4 & 5 \\ 2 & 12 & 13 & 14 & 15 & 16 & 17 \end{bmatrix}$$

$$T_3 \begin{bmatrix} \text{depth} \\ \text{node} \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 3 & 4 & 3 & 2 & 3 & 4 & 5 \\ 3 & 18 & 19 & 25 & 27 & 24 & 28 & 26 & 20 & 21 & 22 & 23 \end{bmatrix}$$

Fig. 3. NDE for trees from Fig. 2

Two NDE operators were developed in order to efficiently manipulate a forest producing a new one. The developed operators can construct NDEs of the modified forests using relatively low running time.

The results produced by the application of both operators (**operator 1** and **operator 2**) are similar. The application of operator 1 (or 2) to a forest is equivalent to the transference of a sub tree from a tree T_{from} to another tree T_{to} of the forest. Applying operator 1, the root of the pruned sub tree T_{from} will be also the root of this sub tree in the new tree T_{to}. On the other hand, applying operator 2, the transferred sub tree will have a new root. It can be any node of the sub tree different from the original root. As results, operator 1 produces simple and small changes in the forest, while operator 2 generates larger and complex alterations.

Operator 1 requires a set of two nodes: the prune node p, which indicates the root of the sub tree to be transferred, and the adjacent node a, which is a node of a tree different from T_{from} and also adjacent to p in graph G. The adjacent nodes to p can be stored in an adjacent list [10].

Operator 2 requires a set of three nodes: the prune node p, adjacent node a, and the new root node r of the sub tree.

A. *Operator 1*

Operator 1 produces a forest F' of a graph G when it is applied to other forest F of G. Results of the operator 1 application are equivalent to transfer a sub tree of a tree T_{from} to another tree T_{to} of the same forest. The root of the pruned sub tree will be also the root in the new tree T_{to}.

The description of operator 1 assumes that two nodes are previously known: the prune node p and the adjacent node a. We can assume that the NDE indices of $p(i_p)$ and $a(i_a)$ are know too.

Fig.4 shows a graph with three trees. The trees T_1 and T_2 were chosen as trees T_{from} and T_{to} respectively, in order to show the application of operator 1. Operator 1 is described by the following steps:

1) Determine the range $(i_p - i_l)$ of indices in T_{from} corresponding to the sub tree rooted at node p. Since we know i_p, we only need to find i_l, that corresponds to the last node's index in the sub tree rooted at the node p. The range $(i_p - i_l)$ corresponds to node p at i_p and the consecutive nodes x in array T_{from} such that $i_x > i_p$ and $d_x > d_p$ (dashed lines in Fig. 4(a)), where d_x is the depth of node x and d_p is the depth of node p;
2) Copy the data in the range $i_p - i_l$ from T_{from} into a temporary array T_{tmp} (containing the data of the sub tree being transferred), see Fig. 4(b). The depth of each node x from the rang $i_p - i_l$ is updated as follows: $d_x = d_x - d_p + d_a + 1$;
3) Create an array T'_{to} and T_{tmp}, inserting T_{tmp} from position i_a of T'_{to} (see Fig. 4(c);
4) Construct an array T'_{from} comprising the nodes of T_{from} without the nodes of T_{tmp} (see Fig. 4(d));
5) Copy the forest data structure F to F' exchanging te pointers arrays T_{from} and T_{to} for pointers to arrays T'_{from} and T'_{to} respectively.

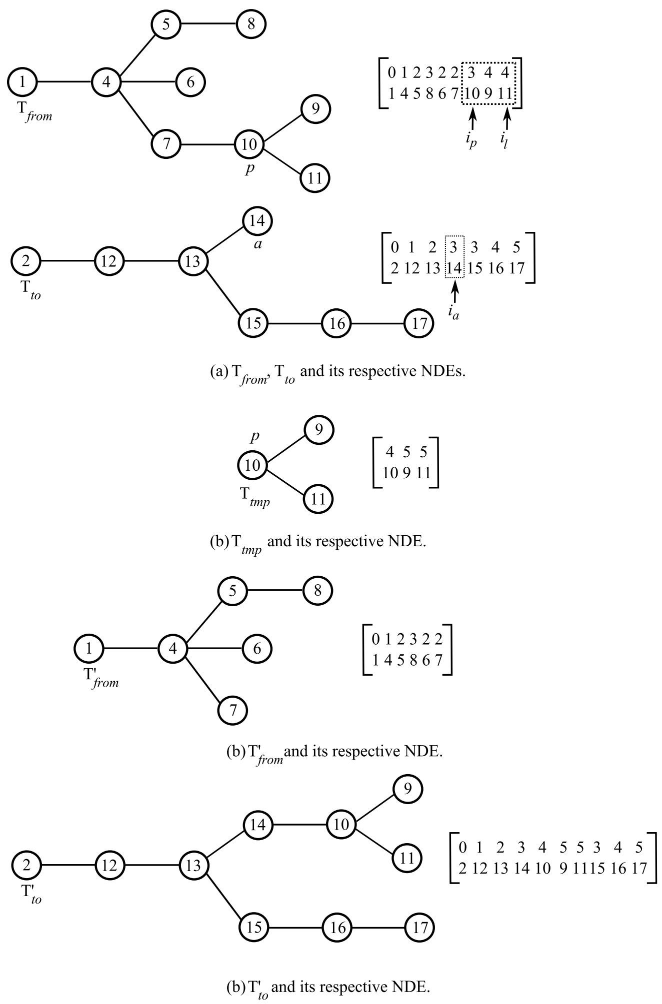


Fig. 4. Example of operator 1 application.

B. Operator 2

Operator 2 also produces a forest F' of a graph G when it is applied to other forest F of G. Results produced by operator 2 are similar to those produced by operator 1, transferring a sub tree from a tree T_{from} to another tree T_{to} of the same forest. However, the sub tree transferred will have a new root which can be any node of the sub tree different of the original root.

The description of operator 2 assumes that a set of three nodes is previously known: the prune node p, the new root node r, and the adjacent node a. Nodes p and r are in the tree T_{from}, and a is in the T_{to}.

The main differences of operator 1 and operador 2 are on the steps 2 and 3 (see Operator 1 procedure), i.e., the formation of pruned sub trees and their storing in temporary array T_{tmp} are different. Next steps 2 and 3 of the operator 2 will be described. Fig. 5. provides an illustrative example of these steps.

The procedure copying the pruned sub tree by operator 2 can be divided into two steps: the first step is similar to step 2 of the operator 1, exchanging i_p for i_r. The second step considers the nodes in the chain from r to p as roots of sub trees (see hithlighted nodes in Fig. 5(a)). In the second step, the sub tree rooted at r_i $(i = 1, .., n)$ witout the subtree rooted at r_{i-1} (5(b)) should be copied and stored in a temporary array T_{tmp} (see Fig. 5(c)).

Operator 2 utilizes the T_{tmp} to construct the tree T'_{to}. The examples illustrated on Fig. 5 considers the trees T_1 and T_2 as the trees T_{from} and T_{to} of the Fig. 2 respectively. The nodes p and r are the nodes 7 and 10 respectively.

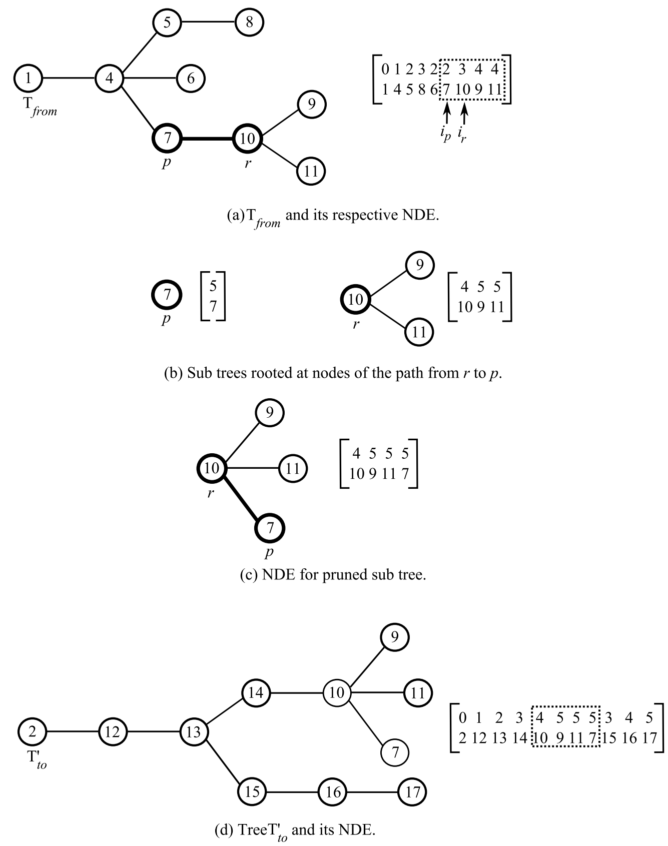


Fig. 5. Determination of T_{tmp} to the operator 2. In (a), the hightlighted lines are the nodes in the chain from r to p. The depths in (b) and (c) are relative to choice of node 14 as adjacent node.

IV. Non-dominated Sorting Genetic Algorithm-II

Such as a conventional EA, NSGA-II also utilizes selection, crossover, and mutation operators to create mating pool and offspring population. However, while conventional EA converts the multi-objective optimization problem into a single objective optimization one with the help of weighting factors, NSGA-II retains the multi-objective nature of the problem.

NSGA-II works with two populations, as in conventional EAs: parents P and offspring Q. In the first iteration, a random parent population P_0 of size N is created and ordered by non-dominance (N is the number of strings or solutions in P_0). Each solution has a fitness according to its nondominance level (1 is the best level, 2 is the second and so on). Then, operators of tournament selection, crossover and mutation are applied to generate the offspring population Q_0 of size N. Both populations P and Q are joined in a population $R_0 = P_0 U Q_0$, with size $2N$. For new iterations, $t = 1, 2, ..., G_max$, the NSGA-II works with R_t populations.

After R_t populations have been obtained, the individuals are ordered by non-dominance in the frontiers F_1, F_2, ..., F_k. The population size is constant, then from $2N$ individuals present in R_t, only N individuals are inserted in the population P_{t+1}; other N are discarded. This insertion must start by the individuals in frontier F_1, following, F_2 and so on.

While $P_{t+1} + |F_i| \leq N$, all individuals in F_i must be inserted in P_{t+1}. If there exist a frontier F_i to be inserted in that $|F_i| > N - P_{t+1}$, the best spread solutions in F_i are chosen by NSGA-II. This spread degree is determined by the method named crowding distance (diversity). All solutions in F_i are ordered according to their decreasing distances. The first $N - |P_{t+1}|$ solutions in F_i are copied for P_{t+1}.

Finally, Q_{t+1} is generated from P_{t+1} by operators of tournament selection, crossover and mutation.

V. Proposed strategy to obtain ERPs

Based on both reference [4] and the NSGA-II, this paper proposes a new strategy to obtain ERPs for large-scale DSs.

The proposed strategy uses a modified NSGA-II together with NDE and its operators proposed in [6].

In general, NSGA-II does not present a good performance when applied to problems involving a large number of objective functions [11]. To deal with this problem, some modifications were made in NSGA-II in order to find ERPs. The proposed mathematical model is shown in Section II (Equation (2)). This model has two objective functions: $e_1(F)$ is the number of switching operations (the crucial objective); and $|\Omega I'(F)|$ is an aggregation function involving voltage drop, power losses and operational constraints. Thus, NSGA-II is capable of minimizing a large number of objectives without incurring into the problem presented in [11]. The generation of populations P and Q is quite different from that realized by NSGA-II. This modification was necessary enable the use of NDE and its operators 1 and 2.

Operator 1 is utilized to connect the out-of-service area to any feeder, generating the first individual of the population. From this first individual, who represents one feasible configuration, other $N-1$ individuals are generated to fill population P_1. The following steps are similar to the NSGA-II presented in Section IV.

Consider G_{max} the maximum generation number. Operator 1 is utilized to generate new individuals for population Q until $G_{max}/2$ has been reached. After the generations counter had reached $G_{max}/2$, Operator 2 is utilized to generate the new population Q. This strategy is adopted to take better advantage of the operators.

Some parameters must be established:

1) Population P and Q, both with size N;
2) Maximum generation number G_{max}.

Algorithm 1 describes the pseudo-code for the proposed NSGA-II with Several objective Functions (NSF).

VI. Simulation Results

The effectiveness of the proposed strategy (NSF) to obtain ERPs for large-scale DS has been studied on a Brazilian DS. This DS is a large-scale DS, from Sao Carlos city, which has 3,860 buses, 635 switches, 3 substations, 23 feeders, 2 transformers of 50MVA and 1 transformer of 25MVA.

Algorithm 1 NSF

Create a population P_1 of N individuals by operator 1
Sorting P_1 by non-dominance
Apply operator 1 to P_1 to create Q_1 of size N
for $t \leftarrow 1, 2, ..., G_{max}$ **do**
    Apply non-dominance algorithm to $R_t = P_t U Q_t$
    $i \leftarrow 1$
    $P_{t+1} \leftarrow \{\}$
    while $|P_{t+1} + \mathcal{F}_i| \leq N$ **do**
        Apply crowding distance algorithm to F_i
        $P_{t+1} \leftarrow P_{t+1} \cup \mathcal{F}_i$
        $i \leftarrow i + 1$
    end while
    Apply crowding distance algorithm to F_i
    Classify F_i by ranking and crowding distance
    Copy the first $N - |P_{t+1}|$ solution of F_i for P_{t+1}
    op ←decide-operator($operator1$, $operaror2$, t)
    $parents$ ← select individuals in P_{t+1} by crowding operator
    Generate a new population Q_{t+1} throught the op in $parents$
    $t \leftarrow t + 1$
end for
$P_{final} \leftarrow P_t$
$Q_{final} \leftarrow Q_t$

The NSF has been programmed using Turion X2 2.0GHz, 2Gb of memory, with Linux operating system, version Ubuntu 8.04, and language compiler C gcc.

The parameters utilized in the simulations were: N = 110 (N = individuals generated in each generation), and Nmax = 200 (maximum number of generations).

The NSF uses two minimization functions: (1) number of switching operations and (2) the aggregation function. This aggregation function is composed of $f(x) = \delta_1 x_1 + \delta_2 x_2 + \delta_3 x_3 + \delta_4 x_4$, where x_1, x_2, x_3 and x_4 are respectively power losses in kW, maximum network loading in p.u., maximum substation loading in p.u., and maximum voltage drop in p.u.; δ_i is the weight of each objective x_i:

$$\delta_1 = 1;$$

$$\delta_2 = \begin{cases} 100, \text{ if, for at least one } j,\ x_j > 1p.u. \\ 0, \text{ otherwise;} \end{cases}$$

$$\delta_3 = \begin{cases} 100, \text{ if, for at least one } i,\ b_i > 1p.u. \\ 0, \text{ otherwise;} \end{cases}$$

$$\delta_4 = \begin{cases} 100, \text{ if, for at least one } i,\ v_i > 0.9p.u. \\ 0, \text{ otherwise.} \end{cases}$$

In order to evaluate the effectiveness of the NSF, it has been assumed a fault in sector 3,668 interrupting the service of the largest feeder in the DS. In all simulations the strategy was able to restore the entire out-of-service area. The Figures 6, 7, 8 and 9 shows the performance of the NSF.

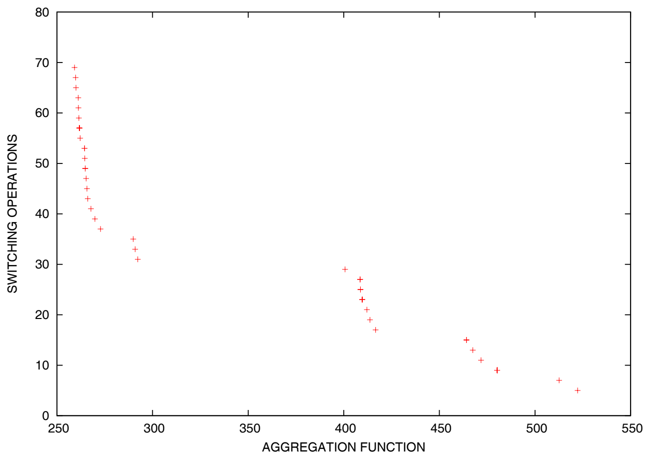


Fig. 6. The best Pareto front which was obtained after the execution of Algorithm 1.

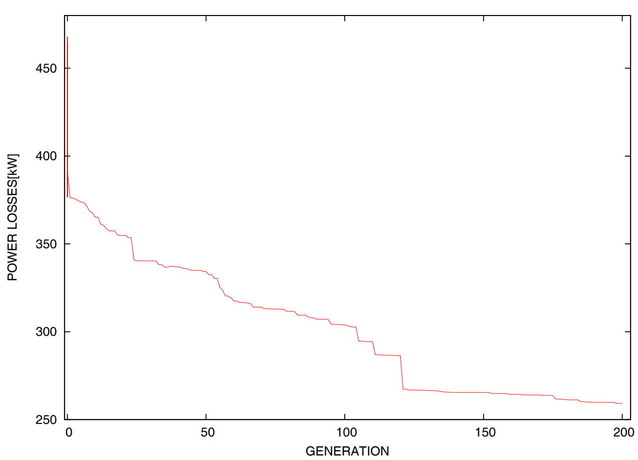


Fig. 7. Power losses in kW of the best individual in each population by generation.

The individual considered as the best is the one having, at the same time, the lowest number of switching operations and the lowest aggregation function value when the process is finished.

TABLE I
SIMULATION RESULTS.

	Initially	Optimum
Total Amount of Power Losses	415,02 kW	355,50 kW
Maximum Voltage Drop	5,02%	4,17%
Maximum Network Loading	139,60%	98,85%
Maximum System Loading	52,72%	52,74%
Number of Switching Operations	21,90	14,8
Processing time	0 s	2.9 s

Table I shows some statistics for 20 runs when sector 3,668 is isolated. Considering that the main objectives of a ERP are (i) restore energy as many loads as possible, (ii) reduced number of switching operations, and (iii) a short running time, the NSF found excellent solutions. Moreover the solutions found by the NSF have a low voltage drop. Configurations with low voltage drop are important in terms of energy service quality.

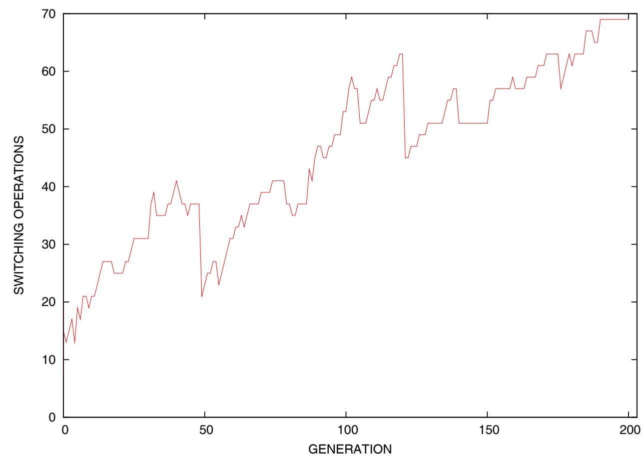


Fig. 8. Switching operations of the best individual by generation.

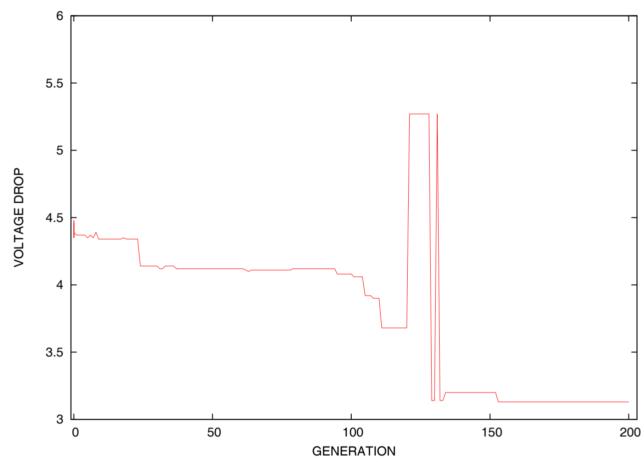


Fig. 9. The maximum drop voltage in pu.

VII. CONCLUSION

The service restoration problem in DS is a combinatorial optimization problem with higher complexity. Several EAs have been proposed over the last decades to deal with this problem. However, the running time may be very long or even prohibitive in applications of EAs to large-scale DS.

In order to overcome this drawback, this paper proposed a new strategy to obtain ERPs for large-scale DS. This strategy modifies a version of the NSGA-II, so that it is able of solving problems with a large number of objectives. Moreover, the proposed strategy utilizes two operators to efficiently manipulate a topology of the system, generating only feasible configurations.

The proposed strategy was applied to the DS of Sao Carlos city in Brazil. Results have demonstrated it enables energy restoration in large-scale DS and solutions were found where: energy was restored to the entire out-of-service area, operational constraints are satisfied, and a reduced number of switching operations is obtained (the most important aspect for the energy restoration problem). Moreover, from the relatively low running time required to elaborate restoration plans for the DS of Sao Carlos city, we can conclude the proposed strategy can elaborate adequate energy restoration plans for large-scale DSs.

ACKNOWLEDGMENT

The authors would like to acknowledge FAPESP for the financial support given to this research.

REFERENCES

[1] A. Augugliaro, L. Dusonchet, and E. R. Sanseverino, "Multiobjective service restoration in distribution networks using an evolutionary approach and fuzzy sets," International Journal of Electrical Power & Energy Systems, vol. 22, pp. 103–110, 2000.

[2] P. Carvalho, L. Ferreira, and L. Barruncho, "On spanning-tree recombination in evolutionary large-scale network problems - application to electrical distribution planning," Evolutionary Computation, IEEE Transactions on, vol. 5, no. 6, pp. 623–630, Dec 2001.

[3] H. Chou, G. Premkumar, and C.-H. Chu, "Genetic algorithms for communications network design - an empirical study of the factors that influence performance," Evolutionary Computation, IEEE Transactions on, vol. 5, no. 3, pp. 236–249, Jun 2001.

[4] A. Santos, A. Delbem, and N. Bretas, "Energy restoration for large-scale distribution system using ea and a new data structure," Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE, pp. 1–8, July 2008.

[5] K. Deb, Multi-objective optimization using evolutionary altorithms. New York: Wiley, 2001.

[6] A. C. B. Delbem, A. C. P. L. F. de Carvalho, C. A. Policastro, A. K. O. Pinto, K. Honda, and A. C. Garcia, "Node-depth encoding for evolutionary algorithms applied to network design." GECCO (1), 2004, pp. 678–687.

[7] Y. Kumar, B. Das, and J. Sharma, "Multiobjective, multiconstraint service restoration of electric power distribution system with priority customers," Power Delivery, IEEE Transactions on, vol. 23, no. 1, pp. 261–270, Jan. 2008.

[8] A. Delbem, A. de Carvalho, and N. Bretas, "Main chain representation for evolutionary algorithms applied to distribution system reconfiguration," Power Systems, IEEE Transactions on, vol. 20, no. 1, pp. 425–436, Feb. 2005.

[9] A. C. Santos, M. Nanni, M. R. Mansour, A. C. B. Delbem, J. B. A. London, and N. G. Bretas, "A power flow method computationally efficient for large-scale distribution systems," Transmission and Distribution Conference and Exposition: Latin America, 2008 IEEE/PES, pp. 1–6, Aug. 2008.

[10] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, Network Flows: Theory, Algorithms,and Applications. Englewood Cliffs: Printce Hall, 1993.

[11] K. Deb and J. Sundar, "Reference point based multi-objective optimization using evolutionary algorithms," in GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation. New York, NY, USA: ACM, 2006, pp. 635–642.

Moussa Reda Mansour (S' 2009) received the B.Sc. degree in computing sciences from the State University of Western Parana, Brazil in 2007.He is Currently pursuing the MSc. degree in electrical engineering at the University of Sao Paulo. His research interest areas are bioinformatics, data structures, graphs, high performance computing, distribution power flow and distribution system reconfiguration.

E-mail: moussa@sel.eesc.usp.br

Augusto Cesar dos Santos (S' 2008) received the B.Sc. degree in electrical engineering from the Federal University of Mato Grosso, Brazil in 2001, and the M.Sc. degree in electrical engineering from the University of Sao Paulo, Brazil in 2004. He is currently pursuing the Ph.D. degree in Electrical engineering in the same institution. He has been a professor at the Federal Technical School of Palmas, Brazil since 2003. His research interest areas are network design, graphs, evolutionary algorithms, and multicriteria problems.

E-mail: augustos@sel.eesc.usp.br

Joao Bosco Augusto London Junior (S'1997-M'02) received the B.S.E.E. degree from Federal University of Mato Grosso, Brazil, in 1994, the M.S.EE in 1997 from E.E.S.C. - University of Sao Paulo, Brazil, and the Ph.D. degree in 2000 from E.P. - University of Sao Paulo, both in Electrical Engineering. He is currently working as an Assistant Professor at the EES.C. - University of Sao Paulo. His main areas of interest are power system state estimation, application of sparsity techniques to power system analysis and distribution system reconfiguration.

E-mail: jbalj@.sel.eesc.usp.br

Alexandre Claudio Botazzo Delbem received the Ph.D. degree from the University of Sao Paulo, Sao Paulo, Brazil, in 2002. He became Assistant Professor at the University of Sao Paulo in 2003. He researches into efficiency in solutions on the real world: large-scale, multicriteria, and realtime. The main areas are power system, scheduling, bioinformatics, and data structures.

E-mail: acbd@icmc.usp.br

Newton Geraldo Bretas received the Ph.D. degree from the University of Missouri, Columbia, in 1981. He became Full Professor at the University of Sao Paulo, Sao Paulo, Brazil in 1989. His work has been concerned with power system analysis, transient stability using direct methods, power system state estimation and energy restoration for distribution systems.

E-mail: ngbretas@sel.eesc.usp.br