MINIMUM NUMBER OF SWITCHING OPERATIONS VIA ANT COLONY OPTIMIZATION

Estéfane George M. de LACERDA
DCA - UFRN – Brazil
E-mail estefane@dca.ufrn.br

Manoel Firmino de MEDEIROS Junior
DCA - UFRN - Brazil
E-mail firmino@dca.ufrn.br

ABSTRACT

This work describes a method for the problem of restoration of electric supply to isolated section of a distribution network. This problem is known as service restoration problem. The restoration is carried out by means of switching operations (i.e., by closing and opening switches) that modifies the network topology such that isolated sections can be feed from alternative sections. An ant colony optimization (ACO) algorithm is proposed in order to minimize number of such switching operations. Experiments with a 130-bus distribution system were carried out.

1 INTRODUCTION

Distribution operators should restore the electric supply as quickly as possible. In practice, they make decisions on the spot, relying on passed experience about switching operations [1]. Because of this the number of operations, is not optimal. Nevertheless, in most distribution systems, the number of switching operations is an essential requirement because the switches are manually operated and the feeders operate close to their capacity.

The problem of minimizing the number of switching operations is a mixed integer nonlinear problem [3] in which there is not a method for finding its exact solution (except for small instances). Metaheuristic algorithms have obtained considerable success in problems of this nature. Metaheuristics as Genetic Algorithm, Tabu Search, and Simulated Annealing have been proposed for service restoration in general [5][6][7]. Some works have presented ACO algorithms for network reconfiguration [8]. Most of them, however, emphasize the optimization of other objectives. For example, minimizing the energy losses and keeping balanced load among feeders. This work presents an ACO algorithm for the specific problem of minimizing the number of switching operations (but it can be adapted to optimize different objectives).

ACO algorithms were inspired from natural behaviour of ant colonies. ACO has been applied successfully to numerous hard optimization problems including the traveling salesman problem (TSP) [2]. Artificial ants are simple agents implementing constructive heuristics. The basic idea of constructive heuristics is incrementally construct solutions by adding, in each step, a solution component to a partial solution until to a complete solution is formed. The cooperation is the key element of ACO algorithms once good solutions are resulted of the cooperative interaction of several artificial ants during the construction of solutions.

A motivation to use ACO algorithms in this work is because already exist, in power system literature, efficient constructive heuristics for network reconfiguration [1][9]. The method, described in [1], starts with all the (tie) switches closed, and in each step, opens a switch. The method stops when the network is radial. In [9], this scheme is inverted. That is to say, the method starts with all the (operable) switches opened, and in each step, closes a switch. Despite their efficiency, these methods are not able of finding the global optima because they are greedy algorithms. The proposed ACO algorithm uses the same switch opening scheme from [1], but it uses a different and inexpensive heuristic information to choice switches. The proposed ACO relies on ant cooperation to guide the switch opening toward promising regions of the search space.

This paper is organized as follows. The optimization problem, tackled in this work, is formulated in Section 2. Section 3 is a short introduction to ACO. The proposed ACO algorithm is presented in Section 4. Section 5 presents experimental results. Finally, Section 6 presents the conclusions and further work.

2 PROBLEM FORMULATION

The service restoration problem consist in minimize n_{ops}, the number of switching operations. That is, minimize $f = n_{\text{ops}}$. By performing switching operations, the network topology is modified and must also satisfy a number of constraints. To deal with these constraints penalty terms are add to objective function. So the optimization problem is

$$\text{minimize } f = n_{\text{ops}} + \lambda_c g_c + \lambda_v g_v \tag{1}$$

where g_c and g_v are penalties associated to violation of current capacity and voltage drop, respectively. λ_c and λ_v are penalty coefficients. The penalties are given by

$$g_c = \sum_{k=1}^{n_{\text{branch}}} \max(|I_k| - I_{(k,\max)}, 0)^2 \quad (2)$$

$$g_v = \sum_{k=1}^{n_{\text{bus}}} \Big[\max(|V_k| - V_{\max}, 0)^2 + \max(V_{\min} - |V_k|, 0)^2\Big] \quad (3)$$

where,

$n_{\text{branch}}, n_{\text{bus}}$	the number of branches and the number of buses of the distribution network
$\|I_k\|, I_{(k,\max)}$	current magnitude and maximum current limit of branch *k*.
$\|V_k\|$,	voltage magnitude at bus *k*.
$V_{\min}, V_{\max}$	minimum and maximum bus voltage limits.

Note that penalty terms grow when the constraints are violated and they are equal to zero where constraints are not violated.

3 ANT COLONY OPTIMIZATION

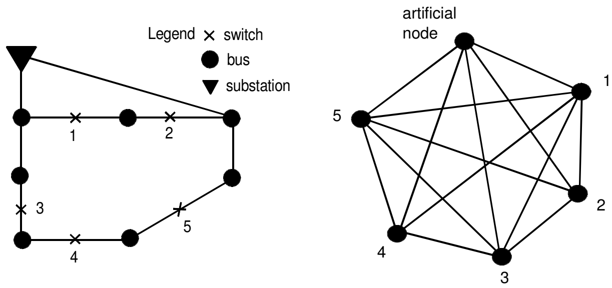
A real ant start the search for food source by wandering randomly until it find the food. After that, it returns to its colony depositing pheromone (a chemical substance) on path *P* between the colony and the food source. If new ants find the path *P*, they follow path *P* with a certain probability according to the level of pheromone left on the path *P*. These ants reinforce the path *P* by depositing more pheromone on *P*. Future ants will choose the path *P* with highest probability due to increase of pheromone level on the path.

O ACO algorithm mimics this behaviour with artificial ants walking around a graph representing solution components of the problem under consideration. By walking around the graph, the ant builds a candidate solution for the problem. That is to say, in each step, the ant walks between two adjacent nodes of the graph and add a solution component to the partial solution. This process finish when the solution is completely built. This graph is commonly called *construction graph*.

In each node of the graph, the ant makes a probabilistic decision in order to choice the next node to go. This decision depends on two types of information: i) the *pheromone level* (representing the desirability of a solution component) and ii) the *heuristic information* (representing a priori information about the problem or runtime information provided by a source different from ants). Just after the ants have built their solution, they return along the trail (i.e., the path) used to construct the solution and deposit the pheromone on it. A basic ACO algorithm is shown in Figure 1.

```
initialize
while termination condition not met do
  foreach ant k = 1,2, ..., m do
    repeat
      select the next node j to be visited with prob. p^k_j (Eq.(4))
    until ant k has built a complete solution
    compute the objective f, Eq.(1), for solution built by ant k
  end for
  update the pheromone
end while
return the best solution found
```

Figure 1 – Ant Colony Optimization



(a) (b)

Figure 2 – (a) Distribution network and (b) the associated construction graph. In ACO, the ants walk on the nodes of the graph. In each node visited the corresponding switch is opened. If an ant, starting at the artificial node, visits node 2 and node 5, then initial meshed network becomes a radial network.

4 AN ACO METHODOLOGY FOR SERVICE RESTORATION

The followings it is the description of each component of the proposed ACO methodology.

Solution Representation. A solution of service restoration problem is represented by a vector of bits $\mathbf{x} = (x_1, x_2, \ldots, x_{ns})$, where n_s is the number of switches of the network. The variable x_k is associated with the switch *k*, which has value one if the switch *k* is closed, and zero otherwise.

Construction Graph. The set of nodes of the graph corresponds to the set of switches (including opened and closed ones) of the network plus an artificial node in which all ants start their tour. The artificial node is necessary in order to specify, without bias, which switch will be opened first. The set of connections of the graph fully connects the set of nodes. In Figure 2 is shown a distribution network with 5 switches and the associated construction graph with 6 nodes (where five of them correspond to switches).

Solution Construction. Initially all the switches of the network are closed creating a meshed network. Next, all ants are put at the artificial starting node. Step by step, each ant iteratively visits a node and opens the corresponding switch. The solution construction ends once the ant has opened a number sufficient of switches. Therefore, after its tour, the ant has built a radial network whose open switches correspond to the visited nodes and the closed switches correspond to no visited nodes.

In each construction step, the ant makes a probabilistic decision in order to choice the next node to go. The ant uses the so-called *random proportional* rule to decide which switch to open. More precisely, the probability p_j of ant k chooses to go to node j is given by

$$p_j^{(k)} = \frac{(\tau_j)^{\alpha}(\eta_j)^{\beta}}{\sum_{k \in N_i^k}^{n} (\tau_k)^{\alpha}(\eta_k)^{\beta}}, \quad if\ j \in N^{(k)} \tag{4}$$

where τ_j is the pheromone level at the node j and η_j is the heuristic information associated to the same node. α and β are scaling factors. The denominator in (4) is a normalizing factor. N^k is the feasible neighborhood of ant k, that is, it is the set of nodes that is allowed to be visited by ant k and is defined in next paragraph.

Constraints. Some constraints are imposed to the walks on the construction graph. Each node is visited at most once. A node cannot be visited if it opens a switch that de-energizes a section of the network. Let L be the number of nodes to be visited. In this work, L is given by

$$L = n_{\text{meshed}} - n_{\text{bus}} + n_{\text{feeder}} \tag{5}$$

Where n_{meshed} is the number of branches of the initial meshed network, n_{bus} is as before and n_{feeder} is the number of feeders of the distribution network. In other words, L is equal to number of switches that should be open to convert the initial meshed network (i.e., the network with all switches closed) into a set of radial feeders.

Pheromone Trails and Heuristic Information. The pheromone is deposited at the nodes of the construction graph (instead of depositing at connections as is done in the TSP problem [2]). So pheromone trails τ_i is associated only with the switch i and refer the desirability of open the switch i. The heuristic information is given by

$$\eta_i = \begin{cases} w, & \text{if } x_i^{(\text{initial})} = 0; \\ 1, & \text{otherwise;} \end{cases} \tag{6}$$

where $x_i^{(\text{initial})}$ indicates the state of the switch i before happening the fault that made the service restoration problem. If $x_i^{(\text{initial})}=0$, then is desirable that switch i be open because it does not generates a switching operation (once it was initially open). The factor $w > 1$ represents the level of this desirability (in this paper $w = 10$).

It worth mentioning that in [1], a switch is selected if its branch has the least current magnitude. Such criterion needs a load flow computation in each construction step. If it is used as heuristic information , it would result in an ACO algorithm with large computational time. Despite its simplicity, the proposed heuristic information in (6) is inexpensive and so it was adopted in this work.

Update of Pheromone Trails. After all ants have built their distribution networks, the pheromone trails are updated. Two events occur in the update: evaporation and depositing of pheromones. The evaporation reduces the level of pheromone on all the switches by a constant value as follows:

$$\tau_j = (1-\rho)\tau_j, \quad \text{for } j = 1, 2, \ldots, n_s \tag{7}$$

where $0 < \rho \leqslant 1$ is the pheromone evaporation rate. The evaporation avoids excessive accumulation of pheromones and so avoids a rapid convergence of the algorithm towards sub-optimal solutions. Furthermore, evaporation is a form of forgetting bad opening switches by previous ants. In the second event, ants deposit pheromone at the switches that they opened as follows:

$$\tau_j = \tau_j + \sum_{k=1}^{m} \Delta\tau_j^{(k)}, \quad \text{for } j = 1, 2, \ldots, n_s \tag{8}$$

where $\Delta\tau_j^{(k)}$ is the amount of pheromone the ant k deposits on the switches it has opened. It is given by:

$$\Delta\tau_j^{(k)} = \begin{cases} 1/f^{(k)}, & \text{if } x_j^{(k)} = 0; \\ 0, & \text{otherwise;} \end{cases} \tag{9}$$

where $x_j^{(k)}=0$ indicates that the ant k opened the switch j and $f^{(k)}$ is the objective function value (see Equation (1)) for the solution found by the ant k.

5 EXPERIMENTAL RESULTS

Experiments were carried with a 130-bus network. The network is shown in Figure 3. It has 5 substations, 77 sectioning switches and 29 tie switches. The ACO algorithm was implemented in Java. The ACO parameters were: 1000 ants, 150 was the maximum number of iterations, α=1, β=2, and ρ = 0.5. A fault happened in branch E01-E02 in which left a large number of sections without electric supply. The global minimum number of switching operations is three. ACO found this solution in 135.3s using Linux with AMD Semprom 2800 processor. More results are in Table 1.

Table 1 – Experimental Results

Branch Fault	Time to find the best solution	Number of switching operations
E01-E02	135s	3
C04-C05	173s	3
A06-A07	80s	1
A09-A010 and E10-E11	85s	2

6 CONCLUSION AND FURTHER WORK

This paper has proposed a methodology to the service restoration problem with the ACO metaheuristic. The methodology uses the sequential switch opening scheme from [1] as a constructive algorithm for ACO. In spite of the proposed ACO use a very simple heuristic information, it reduces significantly the computation time because avoids the intensive computation of load flow of the original scheme of [1]. The Experiments showed that the ACO algorithm solves satisfactorily the service restoration problem, in aceptable time, for a network with 5 substations, 130 buses, 77 sectioning switches and 29 tie switches. Further work should adapt the proposed ACO algorithm to dealing with priority costumers (e.g., hospitals) and to handling other electrical constraints. The sequential switch closing scheme from [9] could also be combined with ACO.

ACKNOWLEDGMENT

The authors would like to thank FAPERN and CNPq for their support.

REFERENCES

[1] D. Shirmohammadi, 1992, "Service restoration in distribution networks via network reconfiguration", *IEEE Transaction on Power Delivery,* vol. 7, n. 2, pp. 952-958.

[2] M. Dorigo and T. Stützle, 2004. *Ant Colony Optimization.* The MIT Press, Cambridge, MA.

[3] K. Aoki; T. Satoh; M. Itoh; H. Kuwabara; M. Kanezashi. "Voltage Drop Constrained Restoration of Supply By Switch Operation in Distribution Systems". *IEEE Transactions on Power Delibery*, Vol. 3, No.3, July 1988.

[4] Toune, S.; Fudo, H.; Genji, T.; Fukuyama, Y.; Nakanishi, Y., "Comparative study of modern heuristic algorithms to service restoration in distribution systems", *IEEE Transactions on Power Delivery*, Volume 17, Issue 1, Jan. 2002 Page(s):173 – 181.

[5] Nara, K; Shiose, A; Kitagawa, M; Ishihara, T. "Implementation of genetic algorithm for distriburion systems loss minimum re-configuration". *Transactions on Power Systems*, vol.7, No 3, August 1992.

[6] Jeon, Young-Jae; Kim Jae-Chul. "Network reconfiguration in radial distribution system using simulated annealing and tabu search." *IEEE*, 2000.

[7] Toune, S.; Fudo, H.; Genji, T.; Fukuyama, Y.; Nakanishi, Y., "Comparative study of modern heuristic algorithms to service restoration in distribution systems", *IEEE Transactions on Power Delivery*. Volume 17, Issue 1, Jan. 2002 Page(s):173 – 181.

[8] Ching-Tzong Sua, Chung-Fu Changb, Ji-Pyng Chiou. "Distribution network reconfiguration for loss reduction by ant colony search algorithm". *Electric Power Systems Research*, 2005.

[9] McDermott, T. E.; Drezga, I.; Broadwater, R. P. A "Heuristic nonlinear constructive method for distribution system reconfiguration". IEEE Transaction on Power System, Vol. 14, No. 2, 1999.

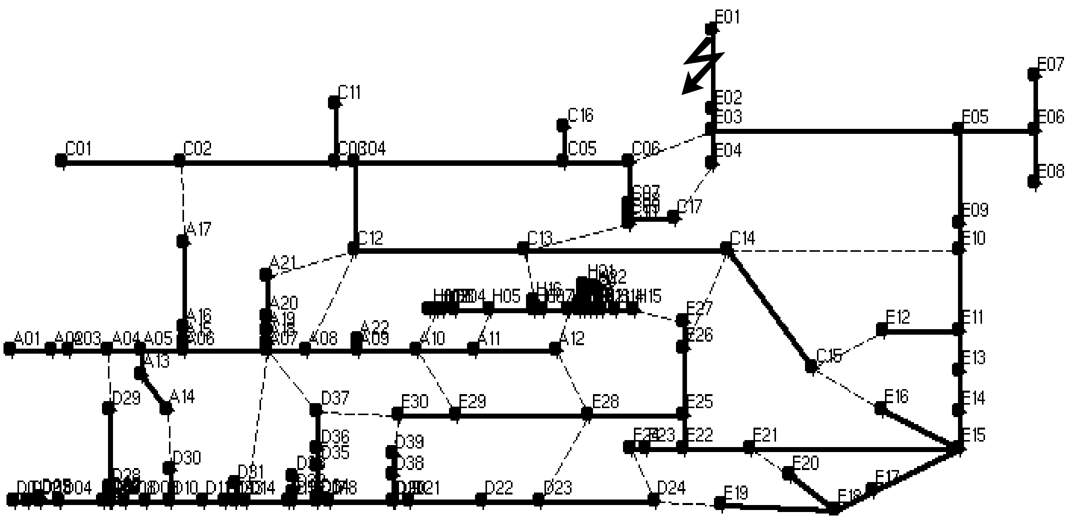


Figure 3 – A 130 bus distribution system. Dashed lines represent tie switches.