*International Journal on Emerging Technologies* **2**(1): 71-77(2011) **ISSN : 0975-8364**

# Development of ACO Algorithm for Service Restoration in Distribution system

***Rajneesh Karn*, Yogendra Kumar* and Gayatri Agnihotri****

**Department of Electrical Engineering, Maulana Azad National Institure of Technology, Bhopal, (MP)*

*(Received 15 January, 2011 Accepted 15 February, 2011)*

**ABSTRACT : Service restoration in power distribution system involves operating the line switches to restore as many loads as possible for the areas isolated by a fault. In case of partial restoration, the supply must be restored to highest priority customers (e.g. hospitals) and this fact should be reflected in the final solution of service restoration problem. Thus service restoration problem is formulated as multi objective, multi constraint combinatorial optimization problem. This paper proposes an Ant Colony Optimization (ACO) algorithm for a minimization problem of energy not supplied during restoration process. The proposed ACO algorithm is a new technique for combinatorial optimization borrowed from swarm intelligence. The operating time of manually controlled and automatically controlled switches is significantly different. Therefore both types of switches are considered separately.**

**Keywords :** Terms-service restoration, Ant colony optimization, Priority customers.

## I. INTRODUCTION

Faulted events are unavoidable in the huge and complex electrical power distribution systems. These faults affect the system's reliability and customer's satisfaction. So the reduction in the effect of fault is necessary to maintain the system's reliability and customer's satisfaction by restoring the service quickly in the area left unsupplied due to fault.

After occurrence of the fault, the operator finds the location of fault, isolate the fault and then restore the service to the healthy components of the out of service area. To meet the service restoration, the alteration of topological structure of distribution system is done by changing the status of switches in distribution system satisfying electrical and structural constraints. For the modern day distribution system, it is hard to implement service restoration solely depending on experimental rules by human operators. To reduce the out of service area as efficiently as possible and the burden of operators, a computer aided decision supports assist the operators. The researchers have developed many methods to solve the service restoration problem in distribution systems [1-12]. Heuristic techniques [1-4] have been developed using heuristic rules. Because heuristic techniques rely on the knowledge of operator of distribution systems, a compromise between knowledge acquisition and performance of the solution is made. Expert systems [5-8] have been developed to quickly determine restoration plans and build look-up tables for distribution personals. Fuzzy set theory [9], network reduction technique [10] and the ranking based search method [11] have also been developed. The petri-net approach [12] has been developed. In [12] the knowledge or the configuration of the concern system in the form of graphic representation is expressed through a structured model. In [11] genetic algorithm is used to solve the service restoration problem.

Before automation in distribution systems, only manual controlled switches were used in distribution system. After development of automatic controlled switches for the purpose of automation, these manual controlled switches started to be replaced by automatic controlled switches. As a result, three categories of presently existing distribution systems can be seen. 1. Those have only manual controlled switches. 2. Those have both manual and automatic controlled switches. 3. Those have automatic controlled switches only. The operating time of both types of switches is different. Therefore, both types of switches should be considered separately.

Some times service restoration for whole out of service area is not possible because power flow in the feeders goes beyond their power transfer capacity. Including the capacitor service controlled action can increase the power transfer capacity of the feeder and, therefore, enhances the chances of full service restoration. If full service restoration is not possible with capacitor control action also, including capacitor control action is definitely helpful to reduce out of service area.

The distribution systems are required to operate in radial fashion for proper relay coordination and ease of fault location etc. So the structure of distribution system should remain radial after service restoration also. In any distribution system, there are always some loads, which are of highest priority (*e.g.* Hospital). In the event of partial service restoration, the supply must be restored to highest priority customers and this fact should be reflected in the final solution of service restoration problem.

In this paper the authors solved the service restoration problem using Ant colony Optimization algorithm. During the optimization process of ACO, there is high probability of generation of better solution at every iteration. The string solution is represented by status of switches. In this work the original configuration and the nearest switch from fault point is considered as starting city of movement for ants. In this method, the following points are considered. 1. Manually controlled switch operation and remotely controlled switch operations are considered separately. 2. Priority customers are considered 3. Out-of service, number of manual switch operation, number of remotely controlled switch operation and losses are minimized. 4. Voltage, current, radiality of the network and supply to priority customers are taken as the constraints. 5. To restore the service to the out of service area, capacitor switches are also considered along with tie switches and sectionalizing switches.

## II. PROBLEM FORMULATION

In this paper, the service restoration problem has been formulated as a multi-objective, multi-constrained combinatorial optimization problem. The various formulations for objective functions and constraints developed in this work are described as follows:

**Objective Functions:**

**1. Minimization of out-of-service area:**

$$\min f_1(\bar{X}) = \sum_{i1}^{b1} L_i - \sum_{i \in B} L_i \qquad \text{... (1)}$$

$\bar{X}$ is switch state vector of network under consideration for service restoration, *i.e.* $\bar{X} = [SW_1, SW_2, .....SW_{N_s}]$

$SW_j$ = Status of $J^{th}$ switch. A closed switch is represented by 1 and an open switch is represented by 0.

$N_S$ = Total number of switches in the network.

b1 = Number of energized buses in the network before fault.

$L_i$ = load on $i^{th}$ bus.

$B$ : Set of energized buses in the restored network.

In equation (1), it is assumed that in a '*n*' bus power system, the buses are numbered from 1 to *n* and in the pre-fault case, all the buses in the network are energized. Therefore '*b*1' is equal to '*n*'. However, in the post fault scenario, all the buses would not be necessarily energized. Hence, 'B' would contain only the energized buses. For example, in a 5 bus system, *b*1 = 5 and if, in the post-fault case, bus 3 can not be energized, then *B* = (1, 2, 4, 5).

**2. Minimization of number of manually controlled switch operation:**

$$\min F_2(\bar{X}) = \sum_{j=1}^{N_m} | SWM_j - SWMR_j | \qquad \text{... (2)}$$

Where, $N_m$ is number of manually controlled switches.

$SWM_j$ = Status of manually controlled switch in network just after fault.

$SWMR_j$ = Status of manually controlled switch in the restored network.

**3. Minimization of number of remotely controlled switch operation:**

$$\min F_3(\bar{X}) = \sum_{j=1}^{N_a} | SWM_j - SWMR_j | \qquad \text{... (3)}$$

Where, $N_a$ is number of remotely controlled switches.

$SWA_j$ = Status of $j^{th}$ remotely controlled switch in network just after fault.

$SWAR_j$ = Status of $j^{th}$ remotely controlled switch in the restored network.

**4. Minimize the losses:**

$\min F_4(\bar{X})$ = Power loss in the restored network which can be calculated with help of load flow. ... (4)

**Constraints:**

1. Radial network structure should be maintained.

2. Bus voltage limits should not be violated.

$$V_{\min} < V_j < V_{\max} \qquad \text{... (5)}$$

$V_{\min}$ = Minimum acceptable bus voltage.

$V_j$ = Voltage at $J^{th}$ bus.

$V_{\max}$ = Maximum acceptable bus voltage.

3. Feeder line current limits should not be violated.

$$I_{\min} < I_j < I_{\max} \qquad \text{... (6)}$$

$I_{\min}$ = minimum acceptable line current.

$I_j$ = Current in $j^{th}$ line.

$I_{\max}$ = maximum acceptable line current.

4. Higher priority customers should always be supplied.

## III. ANT SYSTEM

An Ant Colony Optimization [7, 8, 9, 10] is a random stochastic population based heuristic algorithm on agents that simulate the natural behavior of ants developing mechanisms of cooperation and learning which enables the exploration of the positive feedback between agents as a search mechanism. An important and interesting behavior of ant colonies is their foraging behavior and, in particular, how ants can find shortest paths between food sources and their nest. While walking from food sources to the nest and vice versa, ants deposit on the ground a chemical substance called pheromone, forming in this way a pheromone trail. The sketch shown in the Fig.1. gives a general idea how real ants find a shortest path. Ants can smell pheromone and, when choosing their path, they tend to choose, in probability, paths marked by strong pheromone concentrations. The pheromone trail allows the ants to find their way back to the food by their nest-mates.

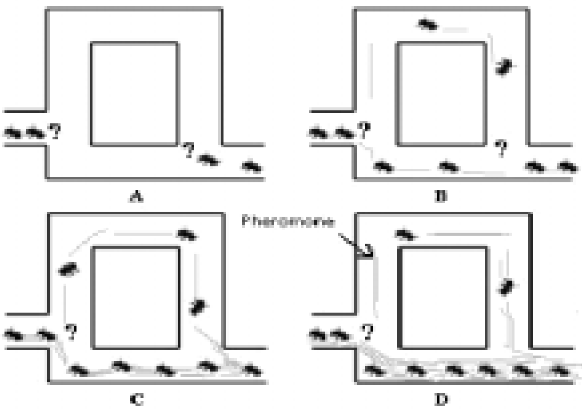


Fig. 1.

The above sketch shows how real ants find a shortest path. *A*. Ants arrive at a decision point. *B*. Some ants choose the upper path and some the lower path. The choice is random. *C*. Since ants move at approximately constant speed, the ants which choose the lower, shorter, path reach the opposite decision point faster than those which choose the upper, longer, path. *D*. Pheromone accumulates at a higher rate on the shorter path. The number of dashed lines is approximately proportional to the amount of pheromone deposited by ants.

## IV. ANT COLONY OPTIMIZATION ALGORITHM

The ant colony optimization problem is very much like traveling sales man problem. In traveling sales man problem(TSP), the salesman has to take the visit of all the towns in such a way that the length of the path remains as minimum as possible. In the same way, the ant has to reach their goal (food) in such a way that the length of the path traveled remains minimum. Therefore, ant colony optimization algorithm is explained below with help of example of traveling salesman problem.

According to the TSP for a given set of n towns, the problem of finding a minimal length of closed tour that visits each town once. We call dij the length of the path between towns *i* and *j*. An instance of the TSP is given by a graph (*N*, *E*), where *N* is the set of towns and *E* is the set of edges between towns. Now, this problem is solved using ant behavior to have the minimum length of the path according following steps called ant colony optimization algorithm.

Let $b_i(t)$ ($i = 1, \ldots, n$) be the number of ants in town $i$ at time $t$ and let $m = b_i(t)$ be the total number of ants.

Each ant is a simple agent with the following characteristics:

* It chooses the town to go with a probability that is a function of the town distance and of the amount of trail present on the connecting edge;

* To force the ant to make legal tours, transitions to already visited towns are disallowed until a tour is completed (this is controlled by a tabu list);

* When it completes a tour, it lays a substance called trail on each edge ($i$, $j$) visited.

Let $\tau_{ij}(t)$ be the intensity of trail on edge ($i$, $j$) at time t. Each ant at time t chooses the next town, where it will be at time $t + 1$. Therefore, if m moves carried out by the m ants in the interval ($t$, $t$ + l) is called an iteration and n is moves required to complete a tour in Ant System, then every $n$ iterations of the algorithm (which we call a cycle) each ant has completed a tour. At this point the trail intensity is updated according to the following formula

$$\tau_{ij}(t+n) = \rho\tau_{ij}(t) + \Delta\tau_{ij} \qquad \text{... (8)}$$

where $\rho$ is a coefficient such that $(1 - \rho)$ represents the evaporation of trail between time $t$ and $t + n$,

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \qquad \text{... (9)}$$

where $\Delta\tau_{ij}$ is the quantity per unit of length of trail substance (pheromone in real ants) laid on edge ($i$, $j$) by the $k^{th}$ ant between time $t$ and $t + n$; it is given by

$$\Delta\tau_{ij}^{k} = \left\{\frac{Q}{L_k}\right\}$$

If ($i$, $j$) $\in$ tour described by $tabu_k$

$$= 0 \quad \text{otherwise} \qquad \text{... (10)}$$

where $Q$ is a constant and $L_k$ is the tour length of the $k^{th}$ ant.

The coefficient $\rho$ must be set to a value < 1 to avoid unlimited accumulation of trail. In order to satisfy the constraint that an ant visits all the $n$ different towns, we associate with each ant a data structure called the $tabu_k$ list that saves the towns already visited up to time $t$ and forbids the ant to visit them again before $n$ iterations (a tour) have been completed. When a tour is completed, the tabu list is used to compute the ant's current solution (*i.e.*, the distance of the path followed by the ant). The $tabu_k$ list is then emptied and the ant is free again to choose. We define tabuk the dynamically growing vector which contains the tabu list, of the $k^{th}$ ant, tabuk the set obtained from the elements of tabuk, and $tabu_k(s)$ the $s^{th}$ element of the list (*i.e.*, the $s^{th}$ town visited by the $k^{th}$ ant in the current tour).

We call visibility $\eta_{ij}$ the quantity $1/d_{ij}$. This quantity is not modified during the run of the AS, as opposed to the trail which instead changes according to the previous formula. We define the transition probability of $k^{th}$ ant to move from town $i$ to town $j$ is given below by equation (11).

$$P_{ij}^{k} = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{k\in\ allowed_k}[\tau_{ik}(t)]^{\alpha}[\eta_{ik}]^{\beta}} \\ 0 \end{cases} \quad (11)$$

where allowed$_k$ = {$N$ – tabu$_k$} and where $\alpha$ and $\beta$ are parameters that control the relative importance of trail versus visibility. Therefore the transition probability is a trade-off between visibility (which says that close towns should be chosen with high probability, thus implementing a greedy constructive heuristic) and trail intensity at time $t$ (that says that if on edge ($i$, $j$) there has been a lot of traffic then it is highly desirable, thus implementing the autocatalytic process). To solve any optimization problem, the value of objective function will be taken in place of $L_k$, $P_{ij}$ is decision parameter to take movement from one state to another state. The stepwise algorithm of the ant colony optimization is given below.

## V. STEPWISE ALGORITHM OF ANT COLONY OPTIMIZATION

The ant-cycle algorithm is simply stated as follows. At time zero an initialization phase takes place during which ants are positioned on different towns and initial values $\tau_{ij}(0)$ for trail intensity are set on edges. The first element of each ant's tabu list is set to be equal to its starting town. Thereafter every ant moves from town $i$ to town $j$ choosing the town to move to with a probability that is a function of two desirability measures. The first, the trail $\tau_{ij}(t)$, gives information about how many ants in the past have chosen that same edge ($i$, $j$); the second, the visibility $\eta_{ij}$ says that the closer a town the more desirable it is. Obviously, setting $a$ = 0, the trail level is no longer considered, and a stochastic greedy algorithm with multiple starting points is obtained. After $n$ iterations all ants have completed a tour, and their tabu$_k$ lists will be full; at this point for each ant $k$ the value of $L_k$ is computed and the values $\Delta\tau_{ij}^{k}$; are updated. Also, the shortest path found by the ants (*i.e.*, mink $L_k$, $k$ = 1, . . . , $m$) is saved and all the tabu lists are emptied. This process is iterated until the tour counter reaches the maximum (user-defined) number of cycles $NC_{MAX}$, or all ants make the same tour. We call this last case stagnation behavior because it denotes a situation in which the algorithm stops searching for alternative solutions.

The ant-cycle algorithm is :

1. Initialize:

Set $t$: : = 0 {$t$ is the time counter}

Set $NC$: : = 0 {NC is the cycles counter}

For every edge ($i$, $j$) set an initial value $\tau_{ij}(t)$ = $c$ for trail intensity and $\Delta\tau_{ij}$= 0, place the m ants on the $n$ nodes.

2. Set $s$ : = 1 {$s$ is the tabu list index }

For $k$: : = 1 to $m$ do

Place the starting town of the kth ant in tabuk(s)

3. Repeat until tabu list is full

{This step will be repeated (n – 1) times}

Set $s$: : = $s$ + 1

For $k$: : = 1 to $m$ do

Choose the town $j$ to move to, with probability

$P_{ij}^{k}(t)$ according to equation (10)

{At time $t$ the $k^{th}$ ant is on town $i$ = tabu$_k$($s$ –1)}

Move the $k^{th}$ ant to the town $j$

Insert town $j$ in tabu$_k$($s$)

4. Move the kth ant from tabu$_k$($n$) to tabu$_k$(1) compute the length $L_k$ of the tour described by the kth ant

Update the shortest tour found

For every edge ($i$, $j$))

For $k$: = 1 to $m$ do

$\Delta\tau_{ij}^{k}$ = {$Q/L_k$} if ($i$, $j$)$\in$ tour described by tabu$_k$

= 0 otherwise

$$\Delta\tau_{ij} = \Delta\tau_{ij} + \Delta\tau_{ij}^{k}$$

5. For every edge ($i$, $j$) compute tij (tth) according to equation

$$\tau(t^{th} = \tau_{ij}(t) + \Delta\tau_{ij}$$

Set $t$ = $t$ + $n$

Set $NC$ = $NC$ +1

For every edge ($i$, $j$) set

6. If ($NC$ < $NC_{max}$) and (not stagnation behavior)

Then

Empty all tabu lists

Goto step 2

Else

Print shortest tour

Stop

## VI. RESULTS AND DISCUSSION

The effectiveness of the Ant Colony System algorithm for service restoration has been studied on two different distribution systems. The details of these systems are given in below. As already mentioned in the previous section, in this work, the Ant Colony technique with reduced run time complexity has been implemented.

In this the data of two test systems namely system-1 and system-2, have been used. The brief summary of each system is given in table-1. The single line diagram, feeder data and bus load data of two systems are as follows

**Table 1.**

| *S. No.* | *Description* | *No. of buses* | *No. of switches* | *Systems nominal voltage (KV)* | *Total system load* | |
|---|---|---|---|---|---|---|
| | | | | | *KW* | *KVAR* |
| 1. | System-1 | 13 | 10 | 11 | 2652 | 866 |
| 2. | System-2 | 10 | 14 | 13.8 | 5600 | 4080 |

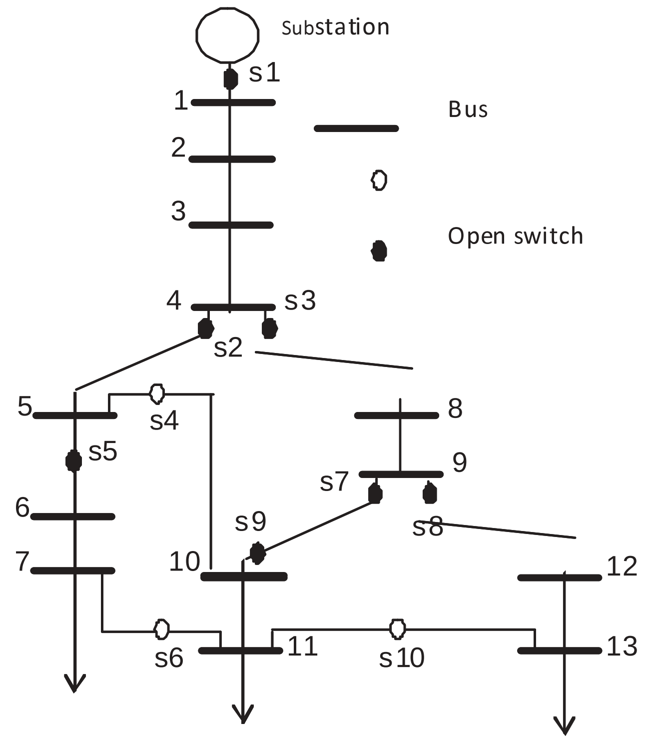


**Table 2: Feeder data of system-1.**

| *Line No.* | *From bus* | *To bus* | *Resistance in ohm* | *Reactance in ohm* |
|---|---|---|---|---|
| 1 | 1 | 2 | 0.148 | 0.287 |
| 2 | 2 | 3 | 0.044 | 0.124 |
| 3 | 3 | 4 | 0.028 | 0.078 |
| 4 | 4 | 8 | 0.160 | 0.310 |
| 5 | 8 | 9 | 0.029 | 0.083 |
| 6 | 9 | 10 | 0.053 | 0.151 |
| 7 | 10 | 11 | 0.059 | 0.166 |
| 8 | 9 | 12 | 0.038 | 0.107 |
| 9 | 12 | 13 | 0.037 | 0.104 |
| 10 | 4 | 5 | 0.060 | 0.167 |
| 11 | 5 | 6 | 0.034 | 0.097 |
| 12 | 6 | 7 | 0.032 | 0.092 |
| 13 | 7 | 11 | 0.047 | 0.101 |
| 14 | 5 | 10 | 0.042 | 0.083 |
| 15 | 11 | 13 | 0.056 | 0.114 |

**Table 3: Bus load data of system-1.**

| *Bus No.* | *Real Power in KW* | *Reactive Power in* |
|---|---|---|
| 1 | 0.000000 | 0.000000 |
| 2 | 473.0000 | 155.0000 |
| 3 | 127.0000 | 41.0000 |
| 4 | 35.0000 | 11.0000 |
| 5 | 438.0000 | 144.0000 |
| 6 | 211.0000 | 69.0000 |
| 7 | 42.0000 | 13.0000 |
| 8 | 473.0000 | 155.0000 |
| 9 | 127.0000 | 41.0000 |
| 10 | 35.0000 | 11.0000 |
| 11 | 438.0000 | 144.0000 |
| 12 | 211.0000 | 69.0000 |
| 13 | 42.0000 | 13.0000 |

**Table 4: Feeder data of system-2.**

| *Line No.* | *From bus* | *To bus* | *Resistance in ohm* | *Reactance in ohm* |
|---|---|---|---|---|
| 1 | 1 | 2 | 0.7820 | 0.2120 |
| 2 | 1 | 3 | 0.7820 | 0.2120 |
| 3 | 1 | 4 | 1.5640 | 0.4240 |
| 4 | 3 | 5 | 1.1730 | 0.3180 |
| 5 | 2 | 6 | 1.1730 | 0.3180 |
| 6 | 3 | 7 | 1.3685 | 0.3710 |
| 7 | 4 | 8 | 1.1730 | 0.3180 |
| 8 | 8 | 9 | 1.1730 | 0.3180 |
| 9 | 2 | 10 | 1.1730 | 0.3180 |
| 10 | 2 | 5 | 0.7820 | 0.3180 |
| 11 | 4 | 10 | 1.1730 | 0.3180 |
| 12 | 3 | 9 | 0.7820 | 0.2120 |
| 13 | 6 | 9 | 0.7820 | 0.2120 |

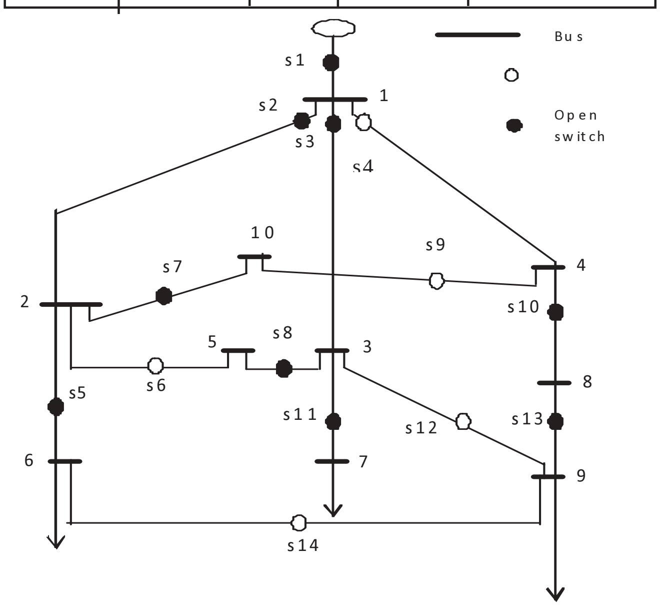

**Table 5: Bus load data of system-2.**

| *Bus No.* | *Real Power in KW* | *Reactive Power in KVAR* |
|---|---|---|
| 1 | 00 | 000 |
| 2 | 600 | 400 |
| 3 | 500 | 300 |
| 4 | 100 | 90 |
| 5 | 600 | 400 |
| 6 | 1300 | 1100 |
| 7 | 1300 | 1000 |
| 8 | 100 | 90 |
| 9 | 800 | 600 |
| 10 | 300 | 100 |

**Table 6: Various single fault location considered and priority customers in two systems.**

| *System* | *System-1* | *System-2* |
|---|---|---|
| Fault location | z4 (between bus 4 and 8) | z6 (between bus 7 and 12) |
| Priority customer | z6 | z3 |

**Table 7: Various multiple fault location considered in all four systems.**

| *System* | *System-1* | *System-2* |
|---|---|---|
| Fault location 1st | z3 (between bus 6 and 7) | z2 (between bus 2 and 3) |
| Fault location 2nd | z4 (between bus 8 and 9) | z6 (between bus 7 and 12) |
| Fault location 3rd | – | z8 (between bus 10 and 11) |
| Fault location 4th | – | – |

**Table 8: Single fault full service restoration.**

| *System* | | *System-1* | *System-2* |
|---|---|---|---|
| Out-of-service area | m1 | 60 | 100 |
| | m2 | 60 | 100 |
| | m3 | 60 | 100 |
| | m4 | 60 | 100 |
| No. of manual Switch operation | m1 | 2 | 1 |
| | m2 | 2 | 1 |
| | m3 | 3 | 2 |
| | m4 | 3 | 2 |
| No. of automatic Switch operation | m1 | 2 | 1 |
| | m2 | 2 | 1 |
| | m3 | 1 | 0 |
| | m4 | 1 | 0 |
| Losses | m1 | 104.50 | 782.40 |
| | m2 | 104.50 | 782.40 |
| | m3 | 106.86 | 779.00 |
| | m4 | 106.86 | 779.00 |
| Run-time of the algorithm | m1 | 15.765 | 18.774 |
| | m2 | 22.538 | 30.121 |
| | m3 | 43.807 | 64.910 |
| | m4 | 13.2040 | 15.180 |
| Status of Priority customer | m1 | y | y |
| | m2 | y | y |
| | m3 | y | y |
| | m4 | y | y |

**Table 9: Single fault partial service restoration.**

| *System* | | *System-1* | *System-2* |
|---|---|---|---|
| Out-of-service area | m1 | 85.3 | 200 |
| | m2 | 85.3 | 200 |
| | m3 | 85.3 | 500 |
| | m4 | 85.3 | 500 |
| No. of manual Switch operation | m1 | 3 | 3 |
| | m2 | 3 | 3 |
| | m3 | 4 | 2 |
| | m4 | 4 | 2 |
| No. of automatic Switch operation | m1 | 2 | 1 |
| | m2 | 2 | 1 |
| | m3 | 0 | 0 |
| | m4 | 0 | 0 |
| Losses | m1 | 77.834 | 778.37 |
| | m2 | 77.834 | 778.37 |
| | m3 | 77.654 | 738.37 |
| | m4 | 77.797 | 733.14 |
| Run-time of the algorithm | m1 | 15.714 | 18.762 |
| | m2 | 21.976 | 29.789 |
| | m3 | 43.782 | 65.117 |
| | m4 | 14.783 | 17.239 |
| Status of priority customer | m1 | y | y |
| | m2 | y | y |
| | m3 | n | y |
| | m4 | n | y |

## VII. CONCLUSION

In this paper, an advanced Ant Optimization technique is developed and is used for solving the service restoration problem in electric power distribution system. In this paper, various practical issues of distribution system operation such as presence of priority customers, presence of remotely controlled as well as manually controlled switches etc. have also been considered. The advantage of the proposed Ant System based technique is that it does not require any operation like crossover & mutation (as needed in a GA based technique). The proposed techniques have many advanced features that are different from other methods:

1. It is easy to deploy and implement to solve many optimal problems,

2. It is very flexible for many problem formulations and

3. It has the ability to escape the local optimal solution and achieve the global optimal solution. Based on a large number of simulation studies it has been found that the Ant System based approach performs better than the GA based approach in solving the service restoration problem.

The main characteristics of this model are positive feedback, distributed computation and the use of a constructive greedy heuristics. Positive feedback accounts for a rapid discovery of good solutions, distributed computation avoids premature convergence and the greedy heuristics helps to find acceptable solutions in the early stages of the search process. Therefore, the Ant System is most suitable for real time implementation (in terms of accuracy and speed both).

## REFERENCES


[1] K. Akoi, H. Kuwabara T. Satoh M. Kanezashi, "Outage state optimal load allocation by automatic sectionalizing switches operation in distribution systems," IEEE trans. Power delivery. 1987,vol. 2, no.4, pp. 1177-1185

[2] K.Akoi , H. Kuwabara T. Satoh M. Kanezashi, "Voltage drop constrained restoration of supply by switch operation in distribution system," IEEE trans. Power delivery. 1988 vol. 3, no. 3, pp. 1267-1279.

[3] K. Akoi, H. Kuwabara T. Satoh M. Kanezashi, "A new algorithm for service restoration in distribution systems," IEEE trans. Power delivery. 1989 vol. 4, no. 3, pp.1832-1839.

[4] C. C. liu S. J. Lee, S.S. Venkata "An expert system operational aid for restoration and loss reduction of distribution systems," IEEE trans.Power syst.1988 vol. 3, no. 2, pp. 619-626

[5] C. E. Lin et al., "A distribution system outage dispatch by data base method with real time revision," *IEEE trans. Power Delivery*, vol. **4.** Jan., (1989).

[6] Imamura et al., "An application of fuzzy reasoning for service restoration" (in Japanese) Trans. IEE Jpn., vol. 4, July 1989.

[7] H. Mori et al., Parallel simulated annealing for power system decomposition," *IEEE trans. Power system* vol. **4**, May, (1994).

[8] S. Toune, H. Fudo, T. Genji, Y. Fukuyama, Y. Nakanishi, "A reactive tabu search for service restoration in electric power distribution system," *IEEE International conference on evolutionary computation,"* Anchorage Alaska, May 4-11, (1998).

[9] Y. Fukuyama and H. D. chaing, "A parallel genetic algorithm for service restoration in electric power systems," in proceeding *IEEE FUZZ/FES Conference*, Yokohama, Japan, Mar., (1995).

[10] A. Augugliaro, L.Dusonchet, E.R. Sanseverino, "Service restoration in compensated distribution networks using a hybrid genetic algorithm," *Electric power syst. Research*, vol. **46**, 59-66, (1998).

[11] W.P. Luan, M.R.Irving, J.S. Daniel, "Genetic algorithm for supply restoration and optimal load shedding in power system distribution networks," *IEE Proc. Gener. Transm. Distrib.*, vol. **149**, no. 2, pp. 145-151, (2002).

[12] Y. T. Hsiao and C. Y. Chien, "Enhancement of restoration service in distribution systems using a combination of fuzzy-GA method," *IEEE.*

[13] Marco Dorigo, Member, *IEEE, Vittorio Maniezzo, and Albert Colorni Ant System*: Optimization by a Colony of Cooperating Agents IEEE Transactions on systems, man & cybernetics-part-b cybernetics, vol 26, no. 1, 29 February, (1996).

[14] Yogendra Kumar, Biswarup Das, Jaydev Sharma, Genetic Algorithm for supply restoration in distribution system with priority customers 9th international Conference on Probabilistic Methods applied to power systems , Sweden, June, (2006).

[15] Y. Kumar, B. Das and J. Sharma, " Multiobjective, multiconstraint service restoration of electric power distribution system with priority customers", *IEEE Transactions on Power delivery*, Vol. **23**, No-1, pp-261-270, Jan., (2008).

[16] M. Dorigo, L.M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Travelling salesman Problem", *IEEE Transactionson Evolutionary Computations*, Vol, **1**, pp.53-66, (1997).

[17] T.stutzle, M.Dorigo, " a short convergence proof for a class of ant colony optimization algorithm", *IEEE Transactions on Evolutionary Computation*, Vol, **6**, Issue: 4, pp. 358-365, (2002).

[18] M.Dorigo and T.Stutzle, "Ant Colony Optimization", Cambridge, MA: MIT Press, (2004).

[19] M. Dorigo, M. Birattari, T. Stutzle "Ant colony optimization: artificial ants as a computational intelligence technique" *IEEE Transactions on Evolutionary Computation*, pp. 358-365, (2006).