

Stochastic search techniques for post-fault restoration of electrical distribution systems

V SUSHEELA DEVI¹ and M NARASIMHA MURTY²

¹Department of Electrical Engineering and ²Department of Computer Science and Automation, Indian Institute of Science, Bangalore 560 012, India
e-mail: mnm@csa.iisc.ernet.in; sheela@ee.iisc.ernet.in

MS received 28 June 1999; revised 12 October 1999

Abstract. Three stochastic search techniques have been used to find the optimal sequence of operations required to restore supply in an electrical distribution system on the occurrence of a fault. The three techniques are the genetic algorithm, simulated annealing and the tabu search. The performance of these techniques has been compared. A large distribution system of over 29 substations, 2500 nodes and 120 feeders has been used.

Keywords. Distribution systems; stochastic search; genetic algorithms; simulated annealing.

1. Introduction

Whenever there is a fault in an electrical distribution system, in order to ensure minimal reduction in system reliability, the areas without supply should be supplied with power as soon as possible. Even though repairing the fault may take a while, it is often possible to quickly restore power to areas isolated by the fault if they can be temporarily connected to some neighbouring feeders which can take the extra load. For this purpose, normally open links between neighbouring parts of the network are generally provided in large distribution systems. Restoration of supply through alternate routes thus basically entails finding a sequence of connections and disconnections of line sections so that all areas of the network are fed. While doing this, care is taken to see that there is no overloading of system components such as transformers, buses, switchgears and line sections. There may be many feasible alternative solutions and it is desirable to find the optimal one. This normally involves finding the route where minimum use of switchgear has to be made.

Different techniques have been suggested in the literature to determine the optimal sequence of operations required for system restoration. Curcic *et al* (1996) present a review of some of the papers. The use of classical integer programming techniques has been reported by Aoki *et al* (1987, 1989). Aoki *et al* (1987) maximise the sections fed, while taking into account the constraints of line capacity and feeder capacity; in the later paper (Aoki *et al* 1989), the voltage drop is also added to the constraints. A knowledge-based approach is investigated by Sakaguchi & Matsumoto (1983), Lee *et al* (1988) and Liu *et al*

(1988). An expert system has been developed in PROLOG (Lee *et al* 1988; Liu *et al* 1988) and the rules are coded using LISP (Sakaguchi & Matsumoto 1983). Search techniques have also been presented (Morelata & Monticelli 1989; Susheela Devi *et al* 1990, 1991, 1995). A binary search tree is used (Morelata & Monticelli 1989) where every switch is set to 0 or 1. Susheela Devi *et al* (1990, 1991, 1995) use a search technique where the search is guided by appropriate heuristics. Time-of-day loading is also considered here, where the load considered is the maximum load for a few hours after the fault occurs, by which time it is likely to be repaired. Curcic *et al* (1997) classify the loads according to their importance into four levels, which play a role if load shedding is required. Hsu & Kuo (1994) use a fuzzy objective function, where the section loading and the number of switchgear operations are represented by membership functions.

All the above techniques are deterministic in nature. The use of stochastic techniques for this problem does not seem to have been adequately investigated. The reasons for this may be the following.

- (1) The standard formulation of stochastic techniques such as the genetic algorithm, simulated annealing technique and tabu search use fixed length strings. Encoding this type of network problem by fixed length strings would lead to extremely long strings which are unwieldy to handle.
- (2) It is the general impression that the time taken by these techniques to produce solutions is too long for practical use.
- (3) Since analysis of these techniques is difficult, the confidence in the solutions generated may not be high enough to justify their use in critical sectors such as the electric power sector.

In order to address these concerns we have experimented with representations for a large distribution system pertaining to a metropolitan city in India. It is to be noted that the time for obtaining a solution for this problem is not so critical and even taking 10–20 minutes for solving it is permissible. In this paper, we report the use of flexible length strings that are compact and easy to handle. The average time taken for any fault is well within limits and the solutions obtained are identical to those obtained by Susheela Devi *et al* (1995).

The three techniques studied have some similar features. All of them have a current candidate solution(s). These solution(s) have to be evaluated and given a figure of merit which is called the fitness function. Some method has to be used to generate the next candidate solution(s) from the present candidate solution. In the genetic algorithm we have operators like selection, crossover, mutation etc. applied on the candidate solutions to yield the next generation of candidate solutions. In the simulated annealing and the tabu search, we talk of generating a neighbour to the present candidate solution.

2. Candidate solutions

All the three stochastic search techniques work with a set of one or more candidate solutions. We first describe briefly the form used to represent candidate solutions (§ 2.1), generated during the search process. The simulated annealing and the tabu search techniques need neighbouring trial solutions to move from one trial solution to another. Section 2.2 gives details of how the neighbouring trial solutions were generated. During the search process, each candidate solution needs to be evaluated for its quality in solving the problem.

Section 2.3 gives the objective function used for evaluating the solutions generated. This objective function is called the fitness function. The representation of the candidate solution and the fitness function were kept the same for all the three stochastic techniques.

2.1 Representation

In the large distribution system considered by us, each node in the network is labelled with a 4-digit or 5-digit number. These numbers are not used in any computation but only serve to uniquely identify each node and the line sections between the nodes. As mentioned earlier, the solution for system restoration is a sequence of operations that specify which line sections need to be open circuited and which are to be energized. It may be expressed in the form

CONNECT	7196	27196
DISCONNECT	5963	1981
CONNECT	5386	8561
DISCONNECT	1879	5607
CONNECT	6555	5606

Such a sequence is represented in the form of a string where each element of the string is the label of a node. The first two elements of the string are interpreted as the line section that needs to be added to the network, the next two elements are then taken as representing the line section to be removed from the network and so on. Since the number of such operations for a solution cannot be fixed *a priori*, the length of the string is left unspecified.

As an illustration of this representation scheme, consider the network shown in figure 1. If there is a fault in section 6–9 and a candidate solution is Connect 10–18, then the string representing this solution is

(10 18).

If a candidate solution were to be Connect 10–18, Disconnect 17–19 and Connect 19–26, then the string representation would be

(10 18 17 19 19 26).

2.2 Finding a neighbouring string

In the simulated annealing and the tabu search technique, at every iteration there is a current trial solution for which a set of one or more neighbouring trial solutions need to be generated. The set of neighbouring trial solutions is generated using the following rules.

- Replacing the first connect section with another available tie section.
- Appending one or more disconnect–connect pairs to the current trial solution.
- Removing one or more disconnect–connect pairs from the current trial solution.
- Replacing one or more disconnect–connect pairs by other disconnect–connect pairs.

The sections connected or disconnected in the formation of any trial solution are always selected at random from a list of line sections that were physically present and available in the actual distribution network.

For example, if the operations below form the current trial solution

CONNECT	6555	5606
DISCONNECT	5963	1981
CONNECT	7196	27196
DISCONNECT	1879	5607
CONNECT	5386	8561,

some of the neighbouring trial solutions could be

(1) CONNECT	6555	5606
(2) CONNECT	6555	5606
DISCONNECT	5963	1981
CONNECT	7196	27196
(3) CONNECT	4509	4934
DISCONNECT	5963	1981
CONNECT	7197	27197
DISCONNECT	1879	5607
CONNECT	5386	8561
(4) CONNECT	4509	4934
DISCONNECT	5247	6472
CONNECT	6474	6601
DISCONNECT	1879	5607
CONNECT	5386	8561
(5) CONNECT	6555	5606
DISCONNECT	2178	4236
CONNECT	6345	4323
DISCONNECT	5963	1981
CONNECT	7196	27196
DISCONNECT	1879	5607
CONNECT	5386	8561.

The neighbouring trial solutions formed may not always be feasible i.e. these operations may produce network configurations with isolated parts or with meshes. For example, in figure 1 where there is a fault in section 6–9 the following trial solution,

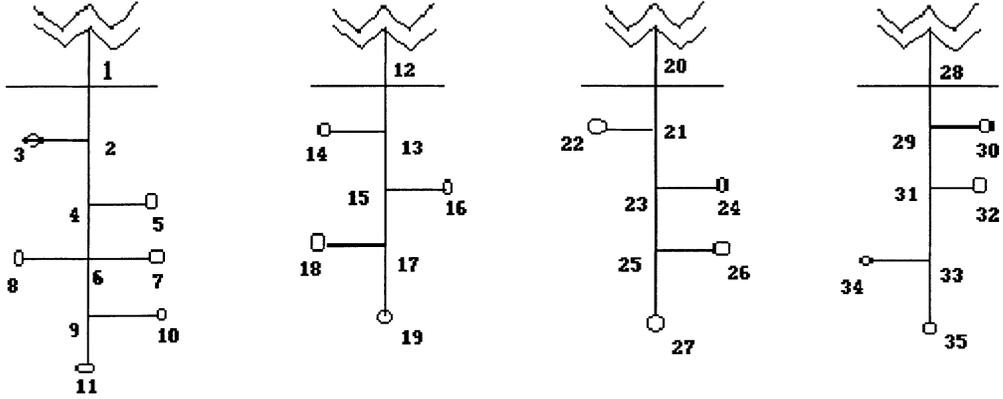
(10 18 23 25 24 35),

produces a network configuration where sections beyond 23–25 are isolated. It also produces a mesh connection where the feeders numbered 20 and 28 feed the same set of nodes.

2.3 Fitness function

The following factors were considered in designing a fitness function suitable for the restoration problem.

- The distribution system must be a radial network.
- All isolated areas must eventually be fed.
- System components must not be overloaded.
- The number of switchgear operations to be made during the restoration process should be minimum.



**Normally Open Sections : 7 - 34 ; 10 - 18 ; 11 - 25 ; 16 - 22 ;
19 - 26 ; 17 - 33 ; 24 - 35 ;**

Figure 1. Sample distribution system.

The first two criteria address the admissibility of a candidate solution and the last two provide a measure of its quality or fitness. Only those networks corresponding to admissible candidates need to be evaluated for the presence of overloaded components and the number of switchgear operations. Hence the fitness function is designed to eliminate the need for load flow computations on networks produced by inadmissible candidate solutions. Keeping in view the requirement that all admissible candidate solutions must have higher fitness values than any inadmissible solution, the fitness function is formulated as given below.

If IS = 0 and MESH = 0 then

$$\begin{cases} \text{if } \text{OVER} > \text{LIM} \text{ then } \text{AMT} = 0 \text{ else } \text{AMT} = 1 - (\text{OVER}/\text{LIM}); \\ \text{fitness} = 1 + \text{AMT} * \text{W1} + [1/(1 + \text{NO})] * (1 - \text{W1}), \end{cases}$$

else

$$\text{fitness} = (1/(1 + \text{IS})) * \text{W2} + ((1/(1 + \text{MESH})) * (1 - \text{W2})),$$

where

IS = number of isolated areas,

MESH = number of meshes,

OVER = amount of overloading,

NO = number of switchgear operations,

W1, W2 \in [0,1] are weights that reflect the relative importance given to the four criteria mentioned earlier,

LIM = amount of overloading permitted until the fault is repaired.

The fitness values computed by the above function lie between 0 and 2. All infeasible solutions have values between 0 and 1, whereas feasible solutions have values between 1 and 2. The higher the fitness value, the better the solution.

3. Techniques used

In this section, we describe the three stochastic techniques, namely the genetic algorithm, simulated annealing and the tabu search technique as implemented by us.

3.1 Genetic algorithms

The basic operating principles of genetic algorithms (Goldberg 1989) are based on the principles of natural evolution. There are many variations of the genetic algorithm but the basic form is the simple genetic algorithm. This algorithm works with a set (or population) of candidate solutions (individuals comprising the population) represented as strings. The initial population consists of N randomly generated individuals where N is the size of the population. At every iteration of the algorithm, the fitness of each individual in the current population is computed. The population is then transformed in stages to yield a new current population for the next iteration. The transformation is usually done in three stages by sequentially applying the following genetic operators: (1) Selection, (2) crossover, and (3) mutation. In the first stage, the selection operator is applied as many times as there are individuals in the population. In this stage every individual is replicated with a probability proportional to its relative fitness in the population. The population of N replicated individuals replaces the original population. In the next stage, the crossover operator is applied with a probability P_c , independent of the individuals to which it is applied. Two individuals (parents) are chosen and combined to produce two new individuals (offspring). The combination is done by choosing at random a cutting point at which each of the parents is divided into two parts; these are exchanged to form the two offspring which replace their parents in the population. In the final stage, the mutation operator changes the values in a randomly chosen location on an individual with a probability P_m . The algorithm terminates after a fixed number of iterations and the best individual generated during the run is taken as the solution.

The genetic algorithm used by us is a variation of the simple genetic algorithm in which two additional stages are incorporated in the transformation process to facilitate the solution of the restoration problem. The two new operators corresponding to these stages are (a) merging, and (b) extension. The application of the merging operator is similar to that of the crossover operator except for the creation of offspring from their parents. Only a single offspring is created by removing the first two elements (corresponding to a connect section) of one parent and appending the resulting substring to the other parent. This offspring replaces one of the parents in the population. The extension operator is used in the same way as the mutation operator except that this operator merely appends a new disconnect–connect pair to the individual. The appended pair is randomly selected from the list of available line sections physically present in the network.

The implementation of the genetic algorithm with the two additional problem specific operators is as follows.

- (1) Set the population size N , the probability of crossover P_c , the probability of mutation P_m , probability of merging P_{merge} , probability of extension P_e , and the maximum number of iterations MAXIT.

- (2) Form the initial current population P of trial solutions and evaluate the fitness value of each individual. Set $K = 0$.
- (3) Set $K = K + 1$.
- (4) Generate $P1$ from P by applying the selection operator.
- (5) Transform $P1$ to $P2$ using the crossover operator.
- (6) Transform $P2$ to $P3$ with the merging operator.
- (7) Transform $P3$ to $P4$ with the extension operator.
- (8) Transform $P4$ to $P5$ by applying the mutation operator.
- (9) Set $P = P5$ and evaluate the fitness value of each individual.
- (10) If $K < \text{MAXIT}$ go to 3.
- (11) Designate the best solution generated so far as the final solution and stop.

The representation of trial solutions and the fitness function used in the algorithm above are as described in §§ 2.1 and 2.3 respectively. Two-point crossover is used where the crossover points are not permitted to fall in the middle of a disconnect–connect pair. Mutation is by replacing the disconnect section of a disconnect–connect pair by another disconnect section.

3.2 Simulated annealing

The simulated annealing approach (Kirkpatrick *et al* 1983) is based on the idea of a process in metallurgy called annealing. The algorithm is started with a random trial solution S_0 to the problem. Each trial solution S_0 has an associated fitness value E_0 , which is computed by the problem-specific objective function. A neighbouring trial solution S_1 is generated from S and E_1 is evaluated. If $E_1 > E_0$ then S_1 is accepted as the current solution, otherwise it is accepted with a probability $\exp(-(E_1 - E_0)/T)$ where T , the temperature, is the control parameter. The process is started with an initial value of T which is slowly reduced. At first, with large values of T , the probability of accepting an inferior solution is high but as T is reduced the probability decreases. The process is run till a prespecified final temperature is reached.

The simulated annealing algorithm is implemented as follows.

- (1) Set the initial temperature T_0 , the final temperature T_f , the cooling rate α and the number of inner loop iterations MAXIT .
- (2) Generate a random initial trial solution S_0 and evaluate its fitness E_0 . Set $K = 0$.
- (3) Set $K = K + 1$.
- (4) Generate a neighbouring trial solution S_1 and evaluate its fitness E_1 .
- (5) If $E_0 > E_1$ then calculate the probability

$$p = \exp(-(E_1 - E_0)/T_0).$$

- (6) If $E_0 < E_1$ or $p > \text{random}(0,1)$ set $S_0 = S_1$ and $E_0 = E_1$.
- (7) If $K > \text{MAXIT}$ then set $T_0 = \alpha * T_0$ and $K = 0$.
- (8) If $T_0 > T_f$ go to step 3, else stop.

The simulated annealing procedure is started with a randomly generated current candidate solution represented as detailed in § 2.1. The fitness values of the candidate solutions generated are calculated with the fitness function given in § 2.3. The method used for determining the neighbouring configuration is described in § 2.2.

3.3 Tabu search

In the tabu search technique (Glover 1990; Al-Sultau 1995), a current solution and a set of its neighbours are maintained. The objective function values of the solutions in this set are computed. The best neighbour is then taken as the next current solution if it is permitted. Once a neighbour is taken as a current solution, it is put into a tabu list. The tabu list is a list of all trial solutions that have been chosen as current solutions in the recent past. At any point, it is not permissible to consider trial solutions existing on the tabu list for selection as the next current solution. Instead, the best neighbouring solution not on the tabu list is chosen as the current solution. A new set of neighbours for the new current solution is generated. This process is repeated for a fixed number of iterations. The best solution generated so far is taken as the final solution.

The tabu search as implemented by us can be described as follows.

- (1) Set the maximum number of iterations (ITMAX), the number of neighbours (NTS) and the tabu list size (MTLS).
- (2) Start with an initial trial solution A_c . Set $K = 0$.
- (3) Set $K = K + 1$.
- (4) Evaluate the fitness J_c of the current solution.
- (5) Generate NTS neighbours of the current solution and evaluate their fitness J_1, J_2, \dots, J_{NTS} .
- (6) Sort the trial solutions in decreasing order of fitness.
- (7) Pick the best neighbour which is not tabu (i.e. the trial solution does not exist in the tabu list) and consider it the new current configuration.
- (8) Put the new current solution in the tabu list. If the tabu list already has MTLS entries, remove the oldest trial solution and include the new current solution.
- (9) If $K < ITMAX$, go to 3.
- (10) Pick the best solution obtained so far as the final solution and stop.

The tabu search is begun with a randomly generated trial solution represented as in § 2.1. The method used for generating the neighbouring trial solutions is as detailed in § 2.2. The fitness value of the trial solutions is calculated with the objective function described in § 2.3.

4. Results

The distribution network used in our experiment comprises over 29 substations, 2500 nodes and 120 feeders with 175 normally open tie sections available. Figure 2 shows a small part of this network. Each of the three stochastic search techniques implemented as described in § 3 is applied for faults at random locations in the network. A typical set of values for the parameters used in the experiment is tabulated below.

Fitness function

W1	:	0.8
W2	:	0.5
LIM	:	2 * (normal feeder peak load)

Genetic algorithm

Population size N	:	50
Probability of crossover P_c	:	0.6

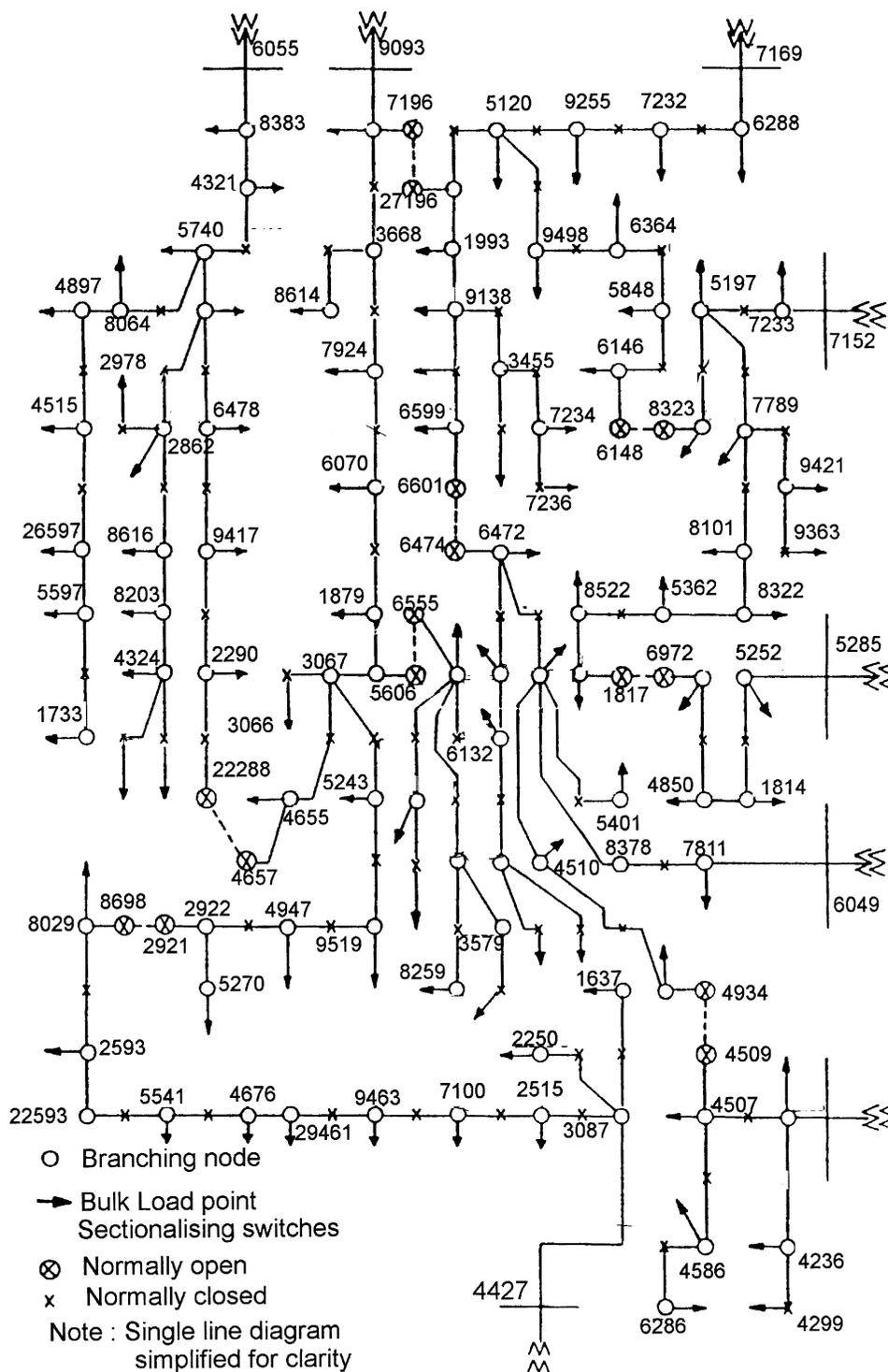


Figure 2. System of 8 feeders.

Table 1. Comparison of time taken using different search techniques.

TS: tabu search; SA: simulated annealing; GA: genetic algorithm

Fault	Time taken (s)		
	TS	SA	GA
7197–9093	39	128	301
2325–5286	67	315	600
6428–4522	18	115	339
6288–7149	55	311	375

Probability of mutation P_m : 0.01
 Probability of merging P_{merge} : 0.15
 Probability of extension P_c : 0.15

Simulated annealing

Initial temperature : 50
 Final temperature T_f : 0.01
 Number of inner loop iterations MAXIT : 10
 Cooling rate α : 0.95

Tabu search

Number of neighbours NTS : 100
 Tabu list length MTL : 100

As mentioned in § 3, the best solution generated during any run is chosen as the optimal solution. The time taken for the generation of the optimal solution for a given fault for ten runs of each technique is recorded. Table 1 presents the average time taken by the three stochastic search techniques for a few fault locations.

From table 1 it is clear that tabu search is faster than the other two stochastic search techniques. Hence additional experiments are made with this technique for more complex faults (that is, requiring more switching operations for restoration). The average time taken to generate the optimal solution is determined as before and tabulated in table 2.

5. Discussion

The representation described in § 2.1 to encode trial solutions is fairly compact and easy to work with even though the algorithm may require a large population or a large number of

Table 2. Time taken using tabu search for a few fault locations.

Fault	No. of switchgear operations	Time taken (s)
7811–6049	5	475
3087–4427	5	438
8383–6055	5	495

neighbours. This method of representation is quite general and may be used for other network problems of this nature.

Regarding the memory requirement, tabu search, which is the fastest of the three techniques, requires the storage of all the neighbours generated. Genetic algorithm also requires a lot of storage since a population of potential solutions has to be maintained at every generation. The simulated annealing technique requires only a single trial solution to be used at every stage and the memory requirement is very little.

An observation made while running any of the stochastic search techniques was the number of infeasible solutions that were generated along the way. These were not suppressed since it was felt that the search process could benefit by using them as stepping stones in the search for better solutions. In any case, the presence of infeasible solutions was not a problem during the search process as the objective or fitness function guiding the search was designed to severely penalise such solutions automatically.

It is very difficult to provide a rigorous mathematical analysis of the pragmatic behaviour of the stochastic techniques described in this paper. These techniques require to be tuned, that is, the values for the parameters need to be set by trial and error. However, they are known to determine, with a high degree of probability, the global optimum in many problem domains. It is interesting to note that all these techniques produce the same solution for any given fault.

Among the three, tabu search obtains the solution faster on an average. However these techniques are amenable to implementation on parallel processors (Moser 1999). While we have not tried it out, we expect this could significantly reduce execution time making it comparable with any deterministic search technique.

The authors wish to acknowledge the contribution of S Sargunaraj in critically going through the earlier drafts of the paper.

References

- Al-Sultan K S 1995 A tabu search approach to the clustering problem. *Pattern Recogn.* 28: 1443–1451
- Aoki K, Kuwabara H, Satoh T, Kanezashi M 1987 Outage state optimal load allocation by automatic sectionalizing switches operation in distribution systems. *IEEE Trans. Power Delivery* PWRD-2: 1177–1185
- Aoki K, Nara K, Itoh M, Satoh T, Kuwabara H 1989 A new algorithm for service restoration in distribution systems. *IEEE Trans. Power Delivery* PWRD-4: 1832–1839
- Curcic S, Ozveren C S, Crowe L, Lo K L 1996 Electric power distribution network restoration: A survey of papers and review of the restoration problem. *Electr. Power Syst. Res.* 35: 73–86
- Curcic S, Ozveren C S, Lo K L 1997 Computer-based strategy for the restoration problem in electric power distribution systems. *Inst. Electr. Eng. Proc.* 144: 389–398
- Glover F 1990 Artificial intelligence, heuristic frameworks and tabu search. *Manag. Decis. Econom.* 2: 365–375
- Goldberg D 1989 *Genetic algorithms in search, optimization and machine learning* (Reading, MA: Addison Wesley)
- Hsu Y, Kuo H 1994 A heuristic based fuzzy reasoning approach for distribution system service restoration. *IEEE Trans. Power Delivery* PWRD-9: 948–953
- Kirkpatrick S, Gelatt C D Jr, Vecchi M P 1983 Optimization by simulated annealing. *Science* 220: 671–680

- Lee S J, Liu C C, Venkata S S 1988 An extended expert system for system restoration of distribution feeders. *IFAC Symposium, Power Systems, Modelling and Control Applications*, pp. 16.4.1–16.4.7
- Liu C C, Lee S J, Venkata S S 1988 An expert system operational aid for restoration and loss reduction of distribution systems. *IEEE Trans. Power Syst.* 3: 619–626
- Morelata A L, Monticelli A 1989 Heuristic search approach to distribution system restoration. *IEEE Trans. Power Delivery* PWRD-4: 2235–2241
- Moser A 1999 *Distributed genetic algorithms for feature selection*. Diplom thesis, University of Kaiserslautern, Germany
- Sakaguchi T, Matsumoto K 1983 Development of a knowledge based system for power system restoration. *IEEE Trans. Power Apparatus Syst.* PAS-102: 320–329
- Susheela Devi V, Sen Gupta D P, Sargunraj S 1990 A search technique for restoring power supply in complex distribution systems, power system for the year 2000 and beyond. *Proc. 6th Nat. Power Syst. Conf.* (New Delhi: Tata McGraw-Hill) 122–125
- Susheela Devi V, Sen Gupta D P, Sargunraj S 1991 Optimal restoration of supply following a fault on large distribution systems. *Int. Conf. Adv. in Power Systems Control, Operation and Management (APSCOM-91)*, Hong Kong
- Susheela Devi V, Sen Gupta D P, Anandalingam G 1995 Optimal restoration of power supply in large distribution systems in developing countries. *IEEE Trans. Power Delivery* 10: 430–438