Proceedings of the Eighth IASTED International Conference
POWER AND ENERGY SYSTEMS
October 24-26, 2005, Marina del Rey, CA, USA

A COMPARATIVE STUDY OF FUZZY CONTROLLED GENETIC ALGORITHMS FOR RECONFIGURATION OF RADIAL DISTRIBUTION SYSTEMS

K. Prasad , N. C. Sahoo
Multimedia University, Bukit Beruang, 75450 Melaka, Malaysia
E-mail: krishna.prasad@mmu.edu.my; Fax: +606-2318840

ABSTRACT: A lot of energy can be saved by efficient reconfiguration of radial distribution systems (RDS) under network constraints. In this paper, a novel genetic algorithm based approach has been adopted for network reconfiguration. The attractive features of the proposed approach are: an improved chromosome coding and an efficient convergence characteristics attributed to fuzzy controlled mutation. A systematic and comparative study of different genetic algorithms and simulation results has been presented and discussed.

KEY WORDS: Radial distribution systems, Network reconfiguration, Genetic algorithm, Fuzzy logic

1. Introduction

RDS reconfiguration alters feeder topology by changing the status of sectionalizing (normally closed) and tie line (normally open) switches. Network reconfiguration is necessary for loss minimization, load transfer from one feeder to other, improving voltage profile and stability. Reconfiguration is obtained by closing a tie-switch and opening a sectionalizing switch to retain the radial topology. The reconfiguration is a complicated combinatorial, non-differentiable and constrained optimization problem owing to the enormous number of switching combinations in distribution systems. Heuristics and expert's experience based approaches can only obtain sub-optimal solutions. Owing to the discontinuous and discrete nature of the problem, classical techniques are rendered unsuitable and the use of global search techniques is warranted.

Evolutionary algorithms, i.e., genetic algorithm (GA) and evolutionary programming (EP) have been proposed for this task [1-4]. The GA is very much suitable for multi-objective optimization. The GA uses a fitness function to guide the search [5],[6]. In [1], a simple chromosome structure represents the switch status in the arcs of the RDS. Thus, the length of chromosome grows with the number of switches. Moreover, there is high probability of crossover operator disrupting the radial nature of the system with a burden to detect and eliminate invalid chromosomes. Song et. al. [2] have used EP by competition and mutation (by a controller) only with an involved data structure. The influence of crossover operator has not been investigated. Further, it becomes necessary to check that the mutation probability lies within the range and it requires special attention, i.e., mutation must be carried out in pairs for preserving the radial property of the RDS. The algorithm proposed in [3] also requires constant examination of the radial structure of the chromosome consuming much time. In [4], a simple mutation controller is used (it is necessary to check that the mutation probability remains within range) and this may not be the best strategy to diversify population for large-scale RDS. In this work, for efficient global search, a fuzzy mutation controller with two inputs, i.e., standard deviation of the population fitness distribution and incremental change in average fitness of population, has been designed. The algorithm has been tested on a standard distribution system.

2. Network Reconfiguration Problem Formulation

The RDS reconfiguration requires determination of a set of branches, possibly one from each loop, to be switched out such that the resulting RDS incurs minimal real loss under network constraints while maintaining the radial nature of the network with all loads being energized. The mathematical statements of optimization and load flow algorithm are discussed below.

2.1 Real Power Loss Minimization

The minimization of real power loss, through reconfiguration, can be stated as:
Minimize kW losses

$$(P_{loss}) = \sum_{m=1}^{N_b} |I_m|^2 R_m \qquad (1)$$

subject to: $V_j^{\min} \le V_j \le V_j^{\max}$, $\forall j = 1,\cdots,N$ and

$|I_m| \le I_m^{max}$, $\forall m = 1,\cdots,N_b$ (2)

where V_j = voltage of node j, I_m = current in branch m, N = number of buses (nodes), N_b = number of branches, and R_m = branch resistance.

2.2 Voltage Profile Improvement

For voltage profile improvement, the following Voltage Deviation Index (*VDI*) is minimized.

$$VDI = \frac{1}{N}\sum_{j=1}^{N}\left|V_j^{sp} - V_j\right| \quad ; \quad V_j^{sp} = \text{specified bus voltage} = 1.0 \text{ p.u} \tag{3}$$

The safe lower limit of the bus voltage is 0.9 p.u.. The *VDI* is a positive quantity under all conditions. It also caters to the situation of the tap-changing transformers/regulators correcting the voltage of the busses closer to the substation and the bus voltage being more than unity.

2.3 Minimization of Current Ratio Index

To minimize branch current violation, the following current ratio index (*CRI*) is minimized.

$$CRI = \frac{1}{N_b}\sum_{m=1}^{N_b}\frac{I_m}{I_m^{max}} \tag{4}$$

3. Fuzzy Controlled Genetic Algorithms for Reconfiguration

The design of the proposed fuzzy controlled genetic algorithm (FCGA) for RDS reconfiguration to minimize losses under network constraints is described. The salient features of the basic genetic algorithm (BGA) are discussed first followed by the specific features of FCGA.

3.1 Features of Basic Genetic Algorithm

BGA is an efficient stochastic search technique in which a group of candidate solutions evolves, through Darwinian principle of natural evolution [5,6], to an optimal solution via the application of genetic operators. The salient features of BGA are [5,6,8]:

- A set of genes, which corresponds to a chromosome, is referred to as a string in BGA.

- BGA starts with a random population of strings and generates successive populations using basic genetic operators, i.e., reproduction, crossover, and mutation.

During reproduction, strings are copied to a "mating pool" using some strategy. From this pool, parents are chosen to mate (crossover) for producing children for the next generation. There are two popular strategies, i.e., roulette wheel selection and tournament selection. There are two popular strategies, i.e., roulette wheel selection and tournament selection.

(a) Roulette Wheel Selection: In this scheme, the individual strings are copied into the mating pool according to their fitness based on 'roulette wheel selection' scheme, i.e., each string occupies an area of the wheel that is equal to the string's share of total fitness. If there are N_p individuals in the population, each having fitness value $f_i, i = 1,2,\cdots,N_p$. Then, the *i-th* individual occupies an area proportional to $f_i \Big/ \sum_{t=1}^{N_p} f_t$. Pairs of strings are picked up randomly from the roulette wheel and each pair undergoes crossover to produce two new strings. There are many types of crossovers, i.e., single point crossover, two-point crossover etc. [37]. It is the most dominant operator in the evolution and its probability (p_c) is high.

(b) Tournament Selection: The binary tournament is the simplest where the fitter of the two randomly chosen strings (with replacement) becomes the first parent. Similarly, the second parent is chosen. Then, the two parents go through crossover to generate children. Mutation is the occasional, with a low probability (p_m), operator used for random alteration of string settings. It diversifies the search and prevents premature convergence. For binary coding scheme, mutation changes a bit from 1 to 0 and vice versa. The genetic operations are repeated until specified maximum generations are reached. Generally, the population size, crossover and mutation probability are chosen empirically.

3.2 Chromosome Coding and Decoding:

In FCGA, the chromosome consists of n binary substrings (n = number of tie lines). Each tie line represents one coded substring, i.e., whether the tie line is closed/open and if it is closed, which branch in the created loop is opened to maintain the radial nature. The chromosome structure is shown in Fig. 2. The leftmost bit of each substring is the status of the tie line (1 ≡ closed, 0 ≡ open). It is followed by some bits to determine the branch selected for opening and these bits have significance if status bit is 1. These bits are decoded to find the branch number by a wrapping-up technique.

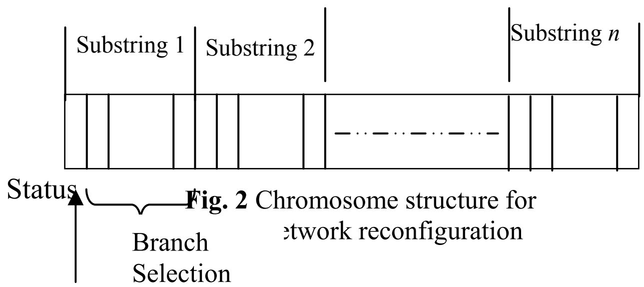

Fig. 2 Chromosome structure for network reconfiguration

In FCGA, the decoding procedure scans the chromosome from left to right. If same branch is encountered more than once, the later choice is ignored by choosing another branch from the loop forming branch set of that tie line and corresponding binary code is inserted to preserve the radial nature. The initial population is randomly chosen with bits being either1 or 0.

3.3 Fitness Function, Reproduction, Crossover and Elitism:

The fitness function should meet the reconfiguration objectives, i.e., minimization of P_{loss}, minimization of *VDI*, and minimization of *CRI*. Thus, the overall fitness function *f* is defined as:

$$f = \frac{1}{1+(\eta_1 \times P_{loss})+(\eta_2 \times VDI)+(\eta_3 \times CRI)} \tag{5}$$

The fitness stays in the range of 0 and 1. η_1 , η_2 and η_3 are chosen constants based on relative weights of the objective (penalty) terms and the FCGA tries to maximize the fitness function. Reproduction and crossover are implemented exactly like BGA. Both roulette wheel and tournament selections are tested with two-point crossover.

3.4 Fuzzy Controlled Mutation*:*

The choice of p_m is critical for good convergence characteristics. Here, a fuzzy system approach is used to compute the mutation probability. The population diversity can be assessed by the standard deviation (σ) of the population fitness distribution and the convergence trend can be evaluated from the incremental change in the average fitness (f_{av}) of the population where f_{av} is defined as:

$$f_{av} = \frac{1}{N_p} \sum_{i=1}^{N_p} f_i \tag{6}$$

A fuzzy rule base (Table 1) with two inputs (σ and Δf_{av}) and one output (p_m) is designed for this purpose. The various fuzzy sets are: LN (Large Negative), SN (Small Negative), ZR (near Zero), SP (Small Positive), and LP (Large Positive), VS (Very Small), S (Small), M (Medium), L (Large), VL (Very Large).

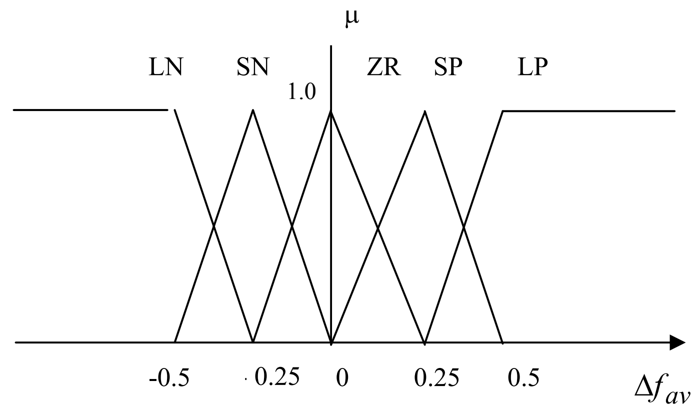

Fig. 3. Membership function for Δf_{av}

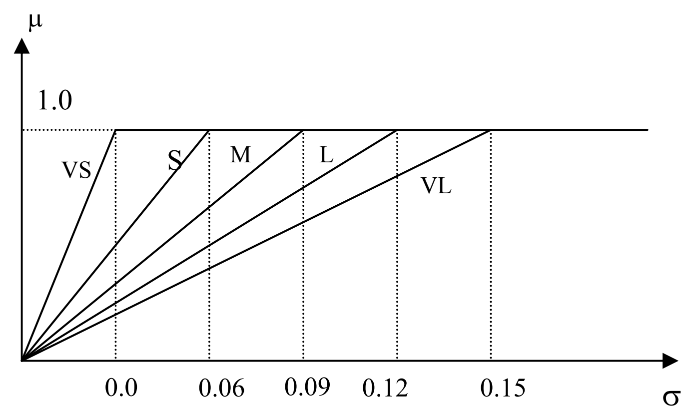

Fig.4. Membership functions for standard deviation

4. Test Results

The proposed FCGA has been tested on a practical 33-bus system [4] The influence of different optimization criterion on the reconfigured network is studied systematically. The fitness function is as in Eqn (5). The central values of fuzzy singletons for the output space of the fuzzy system are chosen as 0.11, 0.12, 0.13, 0.14 and 0.15, respectively, for the different fuzzy sets, i.e., VS, S, M, L, VL. The crossover probability and population size are chosen as 0.9 and 30, respectively.

4.1 Example 1: 33-Bus System

The network reconfiguration objectives are real loss, *VDI* and *CRI* minimization. To visualize the influence of each objective, three case studies have been performed.

4.1.1 Case Study I:

In this case, the fitness function considers only real loss. Thus, simulation tests are done with $\eta_1 = 10$, $\eta_2 = 0$ in Eqn. (5). For two random runs, a comparison of the evolution of maximum fitness of population between roulette-wheel and tournament selection is shown in Fig. 5 where the fixed (complete) fuzzy rule base is used for mutation control. The results show improvement in the utmost performance parameter (Table 2), i.e., real loss of 139.36 kW compared to the base network's loss of 202.67 kW). Further, *VDI* and *CRI* of the reconfigured network have also improved. Performances of both the selection schemes are close to each other. The tournament selection is simple to implement. The roulette-wheel selection may sometimes be affected by stochastic error. Thus, for the subsequent studies, the tournament selection is used.

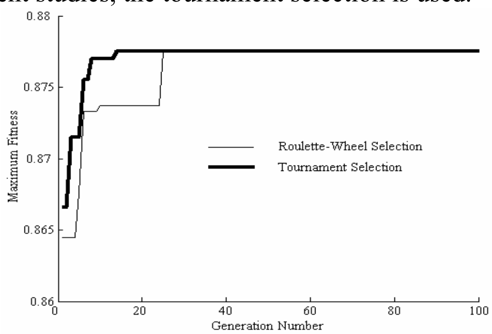

Fig. 5 Performance with different selection schemes; Fixed (complete) fuzzy rule base is used.

As discussed before, the evaluation of all the rules in the rule base may not be efficient. Thus, as described, the evolution of the fuzzy rule base is performed with the auxiliary GA (ε = 0.01). Fig. 6 shows a comparison of performances. The evolved rule base after 200 generations in a random run is shown in Table 2. Here, a simple approach finds the optimal set of rules (out of Table 1) at every generation. The membership function definitions are fixed. For this objective, all the 25 rules in Table 1 are coded by a binary string of 25 bits, i.e., presence (absence) of a rule is coded by 1 (0). For rule base evolution, a population of candidate solutions is used with BGA, in parallel with the evolution for the network reconfiguration using FCGA, based on a trial and error approach. It is observed that although performances are close to each other, the mutation control using evolutionary rule base has a slight edge over the fixed rule base for most runs. In this study, the same best network is obtained as before (Table 2)

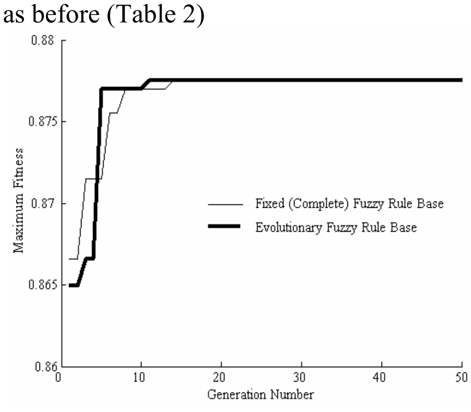

Fig. 6 Performance comparison between fixed and evolutionary rule base mutation controller

The processing (CPU) times (for 100 generations) with different schemes are shown in Table 1. The used computer has a Pentium 4 processor (speed =1.6 GHz). It is seen that the tournament selection with evolutionary rule base has an edge over other schemes. Thus, this strategy is used for rest of the studies. It is emphasized that the actual time taken to get the optimal reconfigured network is much less since this solution is usually obtained within 30 generations.

SCHEME	CPU Time
Roulette-Wheel Selection + Fixed (Complete) Rule Base Mutation	3.30 sec.
Tournament Selection + Fixed (Complete) Rule Base Mutation	3.21 sec.
Tournament Selection + Evolutionary Rule Base Mutation	3.19 sec.

Table 1. Average CPU time for 100 generations with different schemes (33-bus RDS)

4.1.2 Case Study II*:*

Here, the real loss and *VDI* are minimized. The coefficients are: $\eta_1 = 10$, $\eta_2 = 1$. The strategy is tournament selection with evolutionary rule base mutation. The performance measures are reported in Table 2. As a consequence of *VDI* minimization, *VDI* of the reconfigured network is now less than the *VDI* of the network obtained. However, the real power loss is slightly more compared to Case I because, with two objectives to be minimized, FCGA tries to find the optimally best solution.

The reconfiguration by FCGA (Table 2) results in efficient networks compared to base case. The comparison reveals that real loss is the least in Case I, but other parameters have improved in Case II These pictorial comparisons offer better insights for the power system engineer. Finally, it is also necessary compare the performance of FCGA with those of some well-established forms of GA. For this purpose, two algorithms are chosen. They are: (i) Elitist GA [13], and (ii) CHC search algorithm [14]. A brief description of both algorithms follows. In elitist (recombination) GA, there are no separate selection and recombination phases. Crossover is applied to every mating pair. Elitist recombination works at the family level, i.e., every mating pair creates two offspring and the best two of the parents and offspring go to the next generation. Thus, parents are replaced by their own children when the children have higher fitness.

The CHC stands for *Cross-generational elitist selection*, *Heterogeneous recombination* (by incest prevention), and *Cataclysmic mutation*. It is a generational genetic search algorithm with truncation selection mechanism. The parents are randomly paired; but only those string pairs differing from each other by some number of bits (called as mating threshold) are allowed to reproduce. This is the *heterogeneous recombination*. The initial mating threshold is set to one-fourth of the string length. When no offspring are inserted into new population, the mating threshold is decremented by 1. The crossover in CHC performs uniform crossover and randomly swaps exactly half of the differing bits between the parents. After recombination, the next generation population is created by taking the best individuals (number = population size) from the parent and offspring population. This is the *cross-generational elitist selection*. No mutation is applied during the recombination. When no offspring can be inserted into the population of succeeding generation and mating threshold is equal to 0, the CHC introduces fresh diversity into the population via a restart mechanism known as *cataclysmic mutation*. This mutation uses the best individual in the population as a template to re-initialize the population. The new population includes one copy of the template string and the rest of the population is generated by mutating some percentage of bits (e.g. 35% to 40%) in the template string.

A comparative study of performances for random runs is shown in Fig. 7 for Case III. For speeding up the convergence process in CHC, a modification is done and it is in the way the mating threshold is decremented. In addition to the basic CHC, the mating threshold is also decremented by observing the trend of the average fitness of population, i.e., in non-overlapping windows of 10 generations each, if the average fitness is not in increasing trend for at least 5 generations, the mating threshold is decremented.

Fig. 7 also establishes the superiority of FCGA. It is remarked that, in FCGA, the best possible reconfigured network is always obtained within 30 generations and the CPU time required (on Pentium 4 processor) is approximately 1.0 sec. This definitely makes the proposed algorithm a potential candidate for real-time applications.

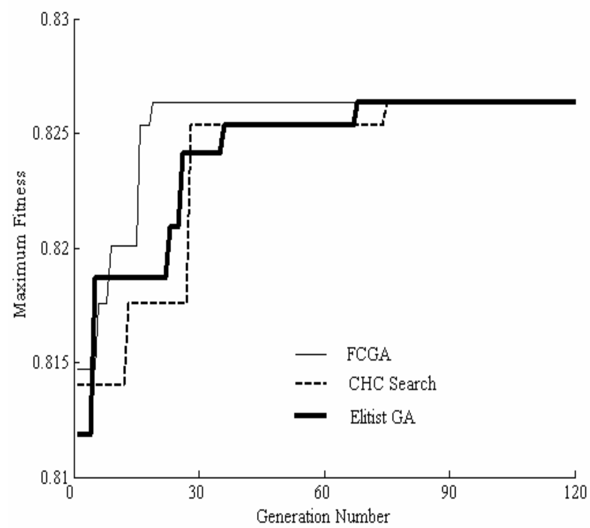

Fig 7. Performance comparisons of FCGA, CHC Search and Elitist GA (for 33-bus RDS)

5. Conclusions

A novel fuzzy controlled genetic algorithm has been proposed in this paper for distribution network reconfiguration minimizing network losses subject to other constraints for overall power quality improvement. A new coding scheme for the chromosome representation of the network has been proposed. Further, a fuzzy logic based mutation controller is proposed for an effective search of the solution space compared to fixed rate mutation operator in BGA. The features of several multi-objective evolutionary algorithms (MOEA) [15], i.e., non-elitist MOEA, Elitist MOEA and constrained MOEA etc. should be investigated along with the proposed FCGA. A study in this direction is presently being pursued.

Cases	Real Loss (kW)	*VDI*	*CRI*	V_{min} (p.u.)	Best Switching Options Branch in-out
Case I	139.36	0.03467	0.04247	0.93782	33-7, 34-14, 35-9, 36-32
Case II	139.782	0.032507	0.03757	0.94129	33-7, 34-14, 35-9, 36-32,37-28
Case III	139.782	0.032507	0.03757	0.94129	33-7, 34-14, 35-9, 36-32,37-28
Base Case	202.67	0.051544	0.05059	0.91308	-----------------

Table 2: Test Results for 33-bus RDS reconfiguration using FCGA

References

1. Nara K., Shiose A., Kitagawa M. and Ishihara T.. (1992), "Implementation of genetic algorithm for distribution systems loss minimum reconfiguration", IEEE Transactions on Power Systems, Vol. 7, No. 3, pp. 1044-1051.
2. Song Y. H., Wang G. S., Johns A. T., and Wang P. Y.. (1997), "Distribution network reconfiguration for loss reduction using fuzzy controlled evolutionary programming", IEE Proceedings - Generation, Transmission, Distribution, Vol. 144, No. 4, July, pp. 345-350.
3. Huang Y. C. (2002), "Enhanced genetic algorithm-based fuzzy multi-objective approach to distribution network reconfiguration", IEE Proceedings - Generation, Transmission, Distribution, Vol. 149, No. 5, September, pp. 615-620.
4. Zhu J. Z.. (2002), "Optimal reconfiguration of electrical distribution network using the refined genetic algorithm", Electric Power Systems Research, Vol. 62, pp. 37-42.
5. Holland J. H. (1975), Adaptation in Nature and Artificial Systems, The University of Michigan Press, Michigan.
6. Goldberg D. E. (1989), Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading.
7. Ranjan R., and Das D. (2003), " Novel computer Algorithm for Solving Radial Distribution Networks", Journal of Electrical Power Components and Systems, Vol.31, No.1, pp.95-107
8. Michalewicz Z. (1996), Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag,.
9. Pedrycz W. (1989), Fuzzy Control and Fuzzy Systems, Wiley, New York.
10. Lee C. C. (1990), "Fuzzy logic in control systems: Fuzzy logic controller – Part I & II", IEEE Transactions on Systems, Man, and Cybernetics,Vol.20, No.2, pp.404 -435.
11. Hoffmann F. (2001), "Evolutionary algorithms for fuzzy control system design", Proceedings of the IEEE, Vol. 89, No. 9, pp. 1318 – 1333.
12. Herrera F., Lozano M., and Verdegay J. L. (1998), "A learning process for fuzzy control rules using genetic algorithms", Fuzzy Sets and Systems Vol.100, pp143– 158.
13. Thierens D. and Goldberg D. (1994), "Elitist recombination: an integrated selection recombination GA", Proceedings of the First IEEE International Conference on Evolutionary Computation, Orlando, USA, Vol. 1, pp. 508 – 512.
14. Eshelman L. J., "The CHC adaptive search algorithm: how to have safe search when engaging in nontraditional genetic recombination", Foundations of Genetic Algorithms, G. Rawlins, ed. Morgan-Kaufmann, Vol. 1, pp. 256 - 283.
15. Deb K. (2001), Multiobjective Optimization Using Evolutionary Algorithms, Chichester, U.K.: Wiley.