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

Radial Network Reconfiguration Method in Distribution System using Mutation Particle Swarm Optimization

T. Sawa, *Member, IEEE*

***Abstract*—We present a new Particle Swarm Optimization (PSO) for optimal radial network reconfiguration. Network reconfiguration is a combinatorial problem. The combination of search space increases exponentially with the number of switches. Fixed loop coding structure is introduced to reduce the number of combinations. The feature of this method is to introduce mutation operation for discrete decimal problem to PSO. This method, discrete decimal PSO with mutation (DDM-PSO), is applied to a test system with 37 nodes and 63 branches. The proposed method improves the maximum arrival number at the optimal solution from 8 to 17 compared to PSO without mutation in twenty trials. This number of DDM-PSO is also more than that of GA, 10. These results show the effectiveness of our method.**

***Index Terms*—combinatorial problem, distribution network, Genetic algorithms, loss minimization, mutation, Particle swarm optimization, radial network, reconfiguration**

I. Introduction

The electric power industry has been extensively deregulated. In Japan ten vertically integrated utilities operate and manage the power system for their regions. Their share of the retail utility market in their regions is more than 96%, but during the early 2000's there was an increase in the share of the retail market held by new suppliers. In recent years fossil fuel prices have greatly increased and global warming has forced utility providers to reduce the levels of their CO_2 emissions. To be competitive under these conditions as well as to establish a "Low carbon society", utilities must reduce their operating costs and improve their energy efficiency. One way to do this is by minimizing the amount of power lost in distribution systems. In fiscal year 2007, transmission and distribution power losses from net generation output to demand were 4.9 % , and 47.6 TWh.

The problem of how to minimize this loss is a large-scale mixed integer-programming one and the related problem of reconfiguration concerns large-scale combinatorial problems. The amount of losses depends on the distribution configuration changed by the status of the section switches. To ensure that a power system can be rapidly restored and to ensure its reliability, distribution systems are operated on a radial configuration. The use of the radial reconfiguration method is the key for a variety of processes involved in the operation of distribution systems: loss minimization, load balancing, and service restoration.

Various reconfiguration methods using meta-heuristics have been proposed to minimize losses. The genetic algorithm (GA) was first proposed by Nara [1] for radial distribution network reconfiguration. There are some coding methods to reconfigure radian networks by using GA. Nara [1] has proposed the use of open switches to do this and Fukuyama, et al [2,3] has proposed the use of upstream nodes. Sawa, et al [4]-[6] has proposed using open switches lined up with an order of loop number, the fixed loop method. This method has been in practical use [7]. Recently particle swarm optimization (PSO) [8] has been considered a realistic and powerful way to search for and obtain the global or quasi-global optimums and use them to optimize power systems [9]. A binary PSO [10] is applied to the radial reconfiguration problem [11].

To find the best way to optimize a radial configuration, we propose using a new method that involves the discrete decimal mutant PSO (DDM-PSO) and the fixed loop method. We applied the method to a test system and show the result here.

II. Coding Structure of individuals

The coding method for the status variable is one of the key issues to apply radial network reconfiguration methods to PSO. An example of a network is shown in Fig. 1. This network contains 9 nodes and 11 branches. Node 1 is a source node. When a network has n nodes and m branches, a necessary condition is for the radial network to open $(m-n+1)$ branches but this is not a sufficient one.

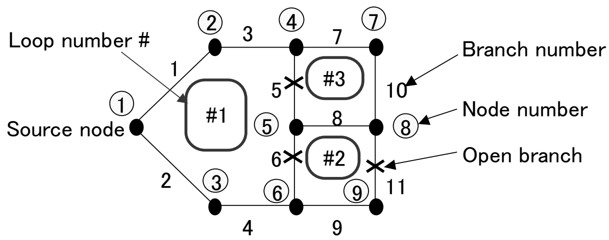

Fig. 1. Example of network.

A. Switch State Method

The most simple topology representation for a network is

T. Sawa is with Hitachi Research Laboratory, Hitachi, Ltd., 7-1-1, Omika-cho, Hitachi, Ibaraki, 319-1292, Japan (e-mail: sawa@ ieee.org).

978-1-4244-2235-7/09/$25.00 ©2009 IEEE

binary coding of each network branch. Usually open branch and closed states are 0 and 1 respectively. This method is used to code an example network as Table I. The dimension of this vector equals the number of branches. The combination number of this method is 2^m and is shown below.

Combination number = 2^{11} = 2048

When this method is applied to GA, a crossover operation is likely to makes infeasible individuals.

TABLE I
Switch state method.

Branch No.	1	2	3	4	5	6	7	8	9	10	11
Switch state	1	1	1	1	0	0	1	1	1	1	0

B. Upstream Node Method

This method uses a decimal coding method shown in Table II. The element of this vector shows an upstream node number in the neighboring node. Node 1 is the power source node such as a generator or substation. Therefore, there is no source node. The second and column number, 1, is a source of node 2. The third column number, 1, is a source of node 3 and so on. The dimension of this vector equals to $(n-1)$. An upstream node is selected from one of nodes in the neighbors. As shown in Fig. 1, 4 nodes have 2 branches and the other 4 nodes have 3 branches in their neighbors.

Combination number = $2^4 \times 3^4 = 1296$

TABLE II
Upstream node method.

Node No.	1	2	3	4	5	6	7	8	9
Upstream node No.	-	1	1	2	8	3	4	7	6

C. Open Branch Method

This method selects $(m-n+1)$ branches to open. This method codes example network as Table 3. The order of the open branch number has no meanings. Therefore, both vectors of 5,6 and, 11 and 6, 11, and 5 have the same meaning in Table III (a) and 3 (b) respectively. The number of $(m-n+1)$ branches is selected from m branches. Therefore, the number of combination is shown below.

Combination number = 11^3 = 1331

TABLE III
Open branch method.

(a)	Open branch No.	5	6	11
(b)	Open branch No.	6	11	5

D. Fixed Loop Method

The "Fixed Loop Method" [3-6] is the improved method from the "Open Branch Method" [1]. In this method, the order of the open branch numbers is important. First $(m-n+1)$ loops are decided in a network on the basis of the following rules. To make all radial networks expressible, every branch must be a member of loops more than once and also, to make one coding data to one network configuration correspondence, be a member no more than twice. Each loop should be made to have fewer branches to minimize the amount of searching space needed. Each loop is assigned a serial number. The number of each open branch is lined up with the order of the loop number.

On the basis of these rules, loops #1, #2, and #3 are made as shown in Fig.1. This method codes the example network as shown in Table IV.

Loops #1, #2, and #3 have 6, 4, and 4 branches respectively, as shown in Table V.

The number of combination is shown below.

Combination number = $6 \times 4 \times 4 = 96$

For the small network in Fig. 1, this number is ten times less than those of other methods [1-3]. For a real distribution network, this difference increases at almost an exponential rate on the basis of the number of branches or nodes. This method is the most effective way to reduce the searching space. When applied to GA, we showed that this method reduced less infeasible solution than the switch state and the open branch method [4].

TABLE IV
Fixed loop method.

Loop No.	1	2	3
Open branch No.	6	11	5

TABLE V
Loop constructed branches.

Loop No.	Loop constructed branches					
#1	1	3	5	6	4	2
#2	6	8	11	9	-	-
#3	5	7	10	8	-	-

III. Implementation of Discrete decimal mutant PSO for Reconfiguration

A. Continuous PSO

Particle swarm optimization is an evolutionary computation technique developed by Kennedy and Eberhart [8]. Its search mechanism is shown in Fig. 3. It is an effective methodology in evolutionary computation that is similar to GA in that the system is initialized with a population of random solutions. In addition, it searches for the optimum by updating generations, and population evolution is based on the previous generations. In PSO, the solutions, called particles, are searched through the problem space by following the current optimal particles. Each particle adjusts its new position on the basis of both its own and its companion's searching experience. The updating velocity of the particles is accomplished by the following (1) that calculates (k+1)-th iteration velocity $\boldsymbol{V}_i^{k+1}$ for each particle based on its previous velocity $\boldsymbol{V}_i^k$, the particle's position $\boldsymbol{pbest}_i$ at which the best fitness so far has been achieved, and the population global position $\boldsymbol{gbest}$ at which the best fitness so far has been achieved. Equation (2) updates each particle's position $\boldsymbol{x}_i^{k+1}$ in the solution hyperspace.

In this way, continuous PSO optimizes the i-th particle's position vector $\boldsymbol{x_i}$ with real numbers. Each vector has n-th dimensional space where n is the size of space of a given problem. For example, the position of i-th particle is described by n-th dimensional vector $\boldsymbol{x}_i^k = (x_{i1}^k, x_{i2}^k, \cdots, x_{in}^k)$, where $\boldsymbol{i} = 1, 2, \cdots, n$.

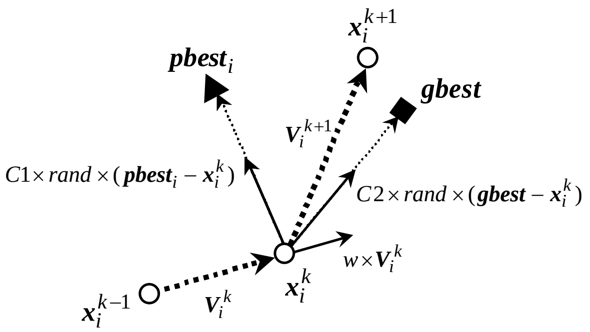

Fig. 2. Searching image of PSO

$$\boldsymbol{V}_i^{k+1} = w \times \boldsymbol{V}_i^k + C_1 \times rand \times (\boldsymbol{pbest}_i - \boldsymbol{x}_i^k) + C_2 \times rand \times (\boldsymbol{gbest} - \boldsymbol{x}_i^k) \tag{1}$$

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{x}_i^k + \boldsymbol{V}_i^k \tag{2}$$

where

$\boldsymbol{V}_i^k$: velocity vector of particle i at generation k;

w : inertia weight;

C_1, C_2 : acceleration constants;

$rand$: random number between 0 and 1;

$\boldsymbol{x}_i^k$: position vector of particle i at generation k;

$\boldsymbol{pbest}_i$: best position vector of particle i until at generation k;

$\boldsymbol{gbest}$: best position vector of the group until at generation k.

B. Bounded Position and Velocity for Particle

The relationship between the real number and the branch number is shown in Fig. 3. The fixed loop method uses decimal variables. Therefore, a real variable is related to a decimal number. The figures and parenthetic figures in Fig. 3 are absolute branch numbers and relative branch numbers respectively. Equations (3), (4), and (5) show how each particle is made to approach to $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$. Equation (3) means that each particle position $\boldsymbol{x}_i$ is bounded from 0 to 2π . As shown in Fig. 4, there are two rotation directions, $\Delta\overline{V}_{ij}$ and $\Delta\underline{V}_{ij}$ for velocity. Equations (4) and (5) mean that each particle deviations from $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$ are set to from $-\pi$ to π to minimize those of absolute values. If this deviation is large, the second and the third terms in (1) are also large and these terms may then make particle position rotate more than one turn. This does not make each particle approach $\boldsymbol{pbest}_i$ and $\boldsymbol{gbest}$.

$$\left.\begin{array}{l} 0 \le x_{ij}^k \le 2\pi \\ \quad if\ \ x_{ij}^k < 0 \qquad then\ \ x_{ij}^k = x_{ij}^k + 2\pi \\ \quad if\ \ x_{ij}^k \ge 2\pi \qquad then\ \ x_{ij}^k = x_{ij}^k - 2\pi \end{array}\right\} \tag{3}$$

$$\left.\begin{array}{l} -\pi \le \Delta V_{ij}^k < \pi \\ \quad \Delta V_{ij}^k = pbest_{ij} - x_{ij}^k \\ \quad\quad if\ \ \Delta V_{ij}^k < -\pi \qquad then\ \ \Delta V_{ij}^k = \Delta V_{ij}^k + \pi \\ \quad\quad if\ \ \Delta V_{ij}^k \ge \pi \qquad then\ \ \Delta V_{ij}^k = \Delta V_{ij}^k - \pi \end{array}\right\} \tag{4}$$

$$\left.\begin{array}{l} -\pi \le \Delta V_{gj} < \pi \\ \quad \Delta V_{gi} = gbest_j - x_{ij}^k \\ \quad\quad if\ \ \Delta v_{gi} < -\pi \qquad then\ \ \Delta v_{gi} = \Delta v_{gi} + \pi \\ \quad\quad if\ \ \Delta v_{gi} \ge \pi \qquad then\ \ \Delta v_{gi} = \Delta v_{gi} - \pi \end{array}\right\} \tag{5}$$

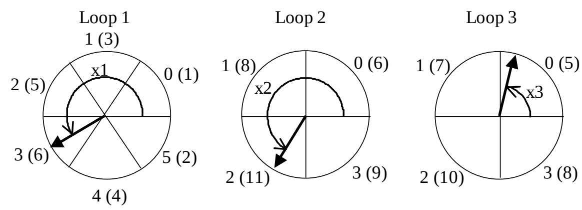

Fig. 3. Relationship between position's real number and branch number.

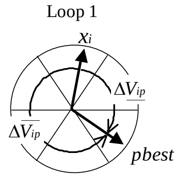

Fig. 4. Difference of velocity for rotation direction.

C. Discrete Decimal Mutant PSO (DDM-PSO) for Reconfiguration

Each particle position is decided by (1) – (5) and each individual makes a radial configuration network. Each open branch in each particle is then mutated if the random number value, $r_{m,}$ is less than the mutation probability P_m. For example, one of open branches is relative No. 3 in Loop 1 and, as shown in (6), a new open branch selects on the basis of a random number value, r_b. Loop 1 has 6 branches and the selection probability for each branch is equal.

$$\begin{aligned}
&if \quad\quad r_b < 1/6 \ then \ No.0 \, branch \ is \ opened, \\
&else \ if \ r_b < 1/3 \ then \ No.1 \, branch \ is \ opened, \\
&else \ if \ r_b < 1/2 \ then \ No.2 \, branch \ is \ opened, \\
&else \ if \ r_b < 2/3 \ then \ No.3 \, branch \ is \ opened, \\
&else \ if \ r_b < 5/6 \ then \ No.4 \, branch \ is \ opened, \\
&else \ if \ r_b < \ 1 \quad then \ No.5 \, branch \ is \ opened.
\end{aligned} \tag{6}$$

IV. Case Studies

Loss minimization, load balancing, and service restoration are problems involved in selecting a radial network that minimizes their objectives. An optimizing structure using PSO and GA is shown in Fig. 5. We selected an objective function on the basis of the problem and considered various constraints. We set the evaluation function for the radial network shown in (7). A penalty was added if constraints were not satisfied for the solutions.

$$\begin{aligned}
&Evaluation \ \ function \\
&\quad\quad = Objective \ \ function + Penalty \ \ function
\end{aligned} \tag{7}$$

We applied this method to the radial reconfiguration problem. This was done to minimize the deviation between the feeder power flow and the feeder capacity. We considered only the radial network constraint. The objective function is shown under for a test network shown in Fig. 6. The penalty function was zero. The simulation condition is shown in Table VI.

$$Objective \ \ function = 1 + \sum_{i=1}^{3} \left| P_i - F_i \right| \rightarrow \min \tag{8}$$

where

P_i : power flow of branch *i*.

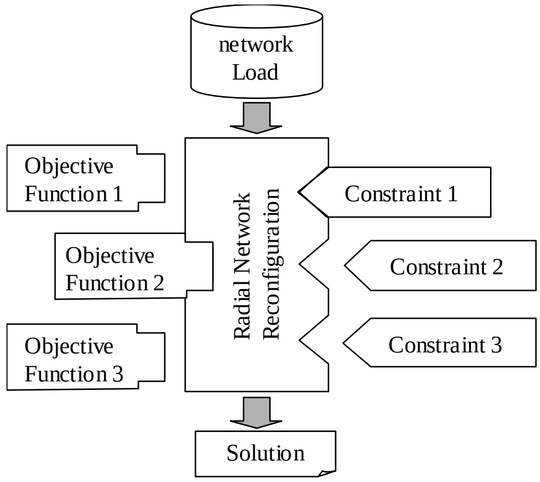

Fig. 5. Radial reconfiguration solving structure using PSO and GA.

The source node was number 0. Feeder capacities, F1, F2, and F3 were 226, 161, and 279 respectively. This test network had 37 nodes and 63 branches. Therefore, a radial network needed to make 27 loops and open one of branches in each loop. The combination number of this test network was 2.0E+16. One of the optimal radial networks is shown below. The optimal solution was to divide the network into the three areas to feed power from feeders F1, F2, and F3. There are various other optimal solutions that used different open branches in the area from Fig. 5. For example an open branch in Loop #3 is changed from branch 9 to 4. The objective function value is one at the optimal solution. That of the second minimum solution, the suboptimal solution, is three. There are three suboptimal solutions based on Fig. 6 as shown in Table VII.

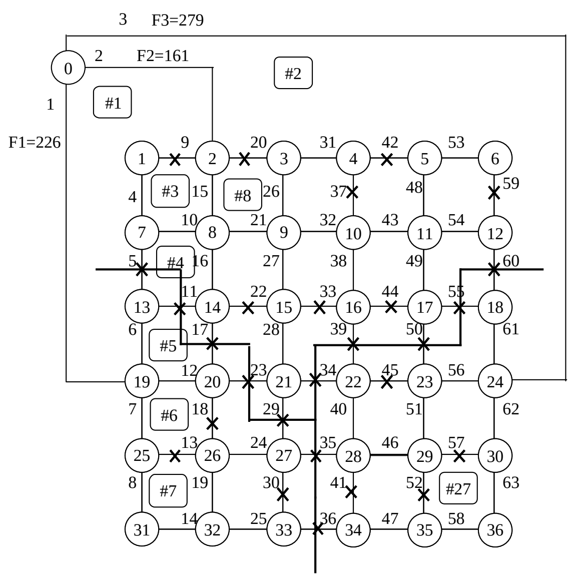

Fig. 6. Test network.

To solve this problem, we applied PSO and GA. The number of individuals is 50. The calculation terminates after the 500th generation. There were twenty trials using different random variables.

TABLE VI
Condition of Simulation

Parameter	Value
Number of particles	50
Number of dimensions	27
Total number of iteration	500
Optimal solution value	0
Suboptimal solution value	1
Number of branches	63
Number of nodes	37

TABLE VII
Suboptimal Solutions

Suboptimal solution	Loop No.	Status changed branches	
		Close	Open
1	#1	5	6
	#4	11	16
	#5	17	11
2	#5	17	12
	#10	23	28
	#11	29	23
3	#2	60	61
	#24	55	49
	#25	50	55

V. Results

The simulation results are shown in Table VIII. The maximum arrival numbers at the optimal solution were 10, 8, and 18 for GA [4], discrete decimal PSO without mutation (DD-PSO) and DDM-PSO respectively. The maximum arrival numbers at more than the suboptimal solution were 18, 11, and 20. The arrival number at the optimal solution is shown in Fig. 8. The mutation probability P_m is 0.1. The inertia weight w is 0.8 for both DD-PSO and DDM-PSO. For DDM-PSO, the arrival number decreases on the basis of the increase in the acceleration constant. On the other hand for DD-PSO, the arrival number increases with the acceleration constant. The average evaluation function value in twenty trials is shown in Fig. 9. This value of DDM-PSO was less than that of DD-PSO in almost all generations.

These results show that using DDM-PSO is more effective than DD-PSO and GA.

TABLE VIII
Simulation Results

	GA	DD-PSO	DDM-PSO
Mutation probability *Pm*	0.6	0.0	0.1
Acceleration constant *C1*	----	2	0.5
Acceleration constant *C2*	----	2	1.5
Inertia weight *w*	----	0.8	0.8
Number of optimal solutions	10	8	17
Number of suboptimal solutions	8	3	3
Number of more than suboptimal solutions	18	11	20

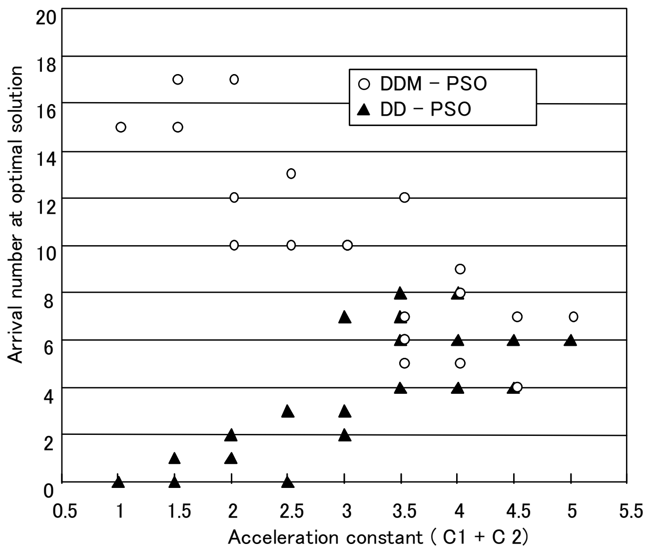

Fig. 8. Relationships between acceleration and optimal arrival number.

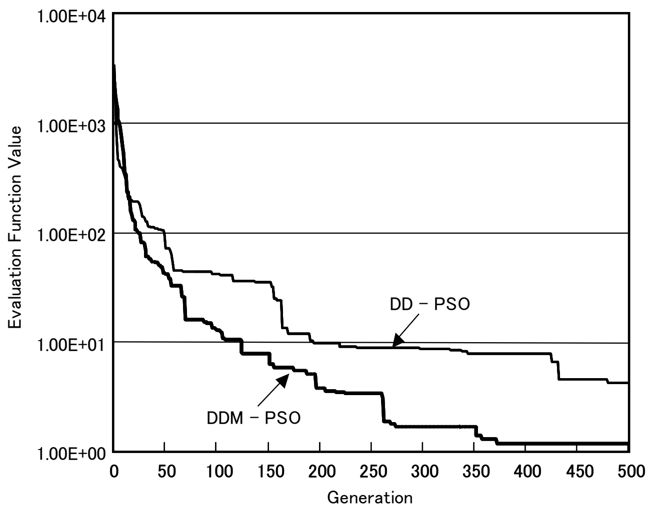

Fig. 9. Changes in generations between DDM-PSO and DD-PSO.

VI. CONCLUSION

We have developed an innovative radial network reconfiguration method using the discrete decimal mutant PSO (DDM-PSO). The feature of this method is to introduce mutation operation and a fixed loop coding structure. This method was applied to a test system with 37 nodes and 63 branches. The proposed method improves the maximum arrival number at the optimal solution from 8 to 17 compared to PSO without mutation. This shows the effectiveness of the developed method.

In the future, we would like to apply loss minimization problems for a distribution network.

VII. REFERENCES

[1] K. Nara, A. Shiose, M. Kitagawa, and T. Ishihara, "Implementation of genetic algorithm for distribution systems loss minimum reconfiguration," *IEEE Trans. Power Syst*., vol.7, no. 3, Aug. 1992, pp. 1044–1051.

[2] S. Toune, H. Fudo, T. Genji, Y. Fukuyama, and Y. Nakanishi, "Comparative study of modern heuristic algorithms to service restoration in distribution systems," *IEEE Trans. on Power Delivery*, vol.17, no. 1, Jan. 2002, pp. 173–181.

[3] M. Kitayama and K. Matsumoto, "An optimization method for distribution system configuration based on Genetic algorithm," *IEEJ Trans. EIS*, vol.116, no. 5, May 1996, pp. 591–598 (in Japanese).

[4] T. Sawa, T. Furukawa, Y. Kobayashi, and S. Tamura, "Radial network configuration method using Genetic Algorithm," *IEEJ Trans. EIS*, vol.121, no. 11, Nov. 2001, pp. 1780–1787 (in Japanese).

[5] W.-M. Lin, F.-S. Cheng and M.-T. Tsay, "Distribution feeder reconfiguration with refined genetic algorithm," *IEE Proceedings in Generation, Transmission and Distribution*, vol. 147, vol. 6, Nov. 2000, pp. 349–354.

[6] J. Mendoza, R. Lopez, D. Morales, E. Lopez, P. Dessante, and R. Moraga, "Minimal loss reconfiguration using genetic algorithms with restricted population and addressed operators: real application," *IEEE Trans. on Power Systems*, vol. 21, no. 2, May 2006, pp. 948–954.

[7] Y. Matsumoto, S. Hase and T. Kinoshita, "Development of outage restoration system using genetic algorithm," *Power Engineering Society Winter Meeting*, vol. 4, Jan. 2000, pp. 2775–2779.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks* (ICNN'95), vol. IV, 1995, pp. 1942-1948.

[9] Y. del Valle, G.K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez and R.G. Harley, "Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems," *IEEE Trans. Evolutionary Computation*, vol. 12, no. 2, Apr. 2008, pp. 171–195.

[10] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 5, 1997, pp. 4104–4108.

[11] R.F. Chang and C.N. Lu, "Feeder reconfiguration for load factor improvement," *Power Engineering Society Winter Meeting*, vol.2, 2002, pp. 980–984.

VIII. BIOGRAPHY

Toshiyuki Sawa (M'97–) was born in Tottori, Japan in 1959. He received the B.S. degree in Mechanical Engineering in 1982 and the M.S. in 1984 from the University of Tokyo, Tokyo, Japan.

After graduation he worked for Hitachi, Ltd. His current research interests include energy management systems, power market systems, and optimization methods. He is a Senior Research Engineer at Hitachi Research Laboratory, Hitachi, Ltd. He is a part-time teacher at Ibaraki University in Japan.

Mr. Sawa is a member of IEEE and the Institute of Electrical Engineers of Japan.