Power Flow Solution Using an Improved Fitness Function in Genetic Algorithms

Seungchan Chang, Jae-Yoon Lim, and Jung-Hoon Kim

Abstract

This paper presents a methodology of improving a conventional numerical model in power systems using Genetic Algorithms(GAs) and suggests a GAs-based model which can directly solve the real-valued optimum in an optimization procedure. In applying GAs to the power flow, a new fitness mapping method is proposed using the probability distribution function for all the payoffs in the population pool. In this approach, both the notions on a way of the genetic representations and a realization of the genetic operators are fully discussed to evaluate the GAs' effectiveness. The proposed method is applied to *IEEE* 5-bus, 14-bus and 25-bus systems and, the results of computational experiments suggest a direct applicability of GAs to more complicated power system problems even if they contain nonlinear algebraic equations.

I. Introduction

Genetic Algorithms(GAs) based on the mechanisms of evolution and natural genetics have strong points in not using derivative informations in an optimization procedure and, they provide robust search in complex spaces and an alternative to traditional optimization techniques by using directed random searches to locate optimal solutions. The main operations of GAs are made by reproduction or selection, crossover, mutation process and, their carrying major factors are population size, chromosome length, crossover rate, mutation rate and so on. According to the objective function how to construct, we need to prescribe its adequate fitness function. And, a variety of encoding, crossover and mutation methodologies enable us to have diverse approaches to the problems[1, 2].

Over the last decade, considerable research has focused on improving GAs performance[1]. Efficient implementations ^{0} of the proportionate selection scheme such as the stochastic remainder technique and the stochastic universal sampling technique have been proposed to reduce the sampling errors. Reproduction mechanisms such as rank-based selection, elitist strategies, steady state selection, and tournament selection have been proposed as alternatives to the proportionate selection scheme. Crossover mechanisms such

as two-point, multi-point and uniform crossover have been proposed as improvements on the traditional single point crossover technique. Gray codes and dynamic encoding have overcome some problems associated with fixed point integer encoding. Departing from the traditional policy of static control parameters for the GAs, adaptive techniques dynamically vary the genetic control parameters. Recently, significant innovations include the distributed GAs and parallel GAs[2].

Most of the power system analysis takes on highly nonlinear and computationally difficult characteristics in an optimization procedure. To solve these problems, many methodologies have been devised and modified so far. In recent years, there has been a growing concern for GAs applications to the power system problems. Many papers have established the validity of GAs applicability to the power system control and operation such as economic dispatches[3], reactive power optimization[4], thermal unit commitment[5], distribution network planning[6] and so on. But, most of these works take GAs as a pre-searching tool in the optimization procedure and so stick to use a hybrid-type model.

In 1991, Xiaodong Yin and Noël Germay[7] experienced GAs applicability to the load flow problem with Klos-Kerner 3-node system and Ward-Hale 6-node system using the concept of Holland's SGAs. Before applying SGAs to the load flow, some factors about the genetic representation have to be considered such as, coding the variables into a finite string, treatment of constraints and mapping the objective function to a fitness function. Predefined fitness function which is an inversed type of objective is evaluated during the entire genetic process. To enhance the process

property, it uses a sharing function that the population is divided into different subpopulations according to the similarity of the individuals in the population. This approach will have a good convergence especially when the variables of representation are small, but the fitness function itself may consecutively accumulates erroneous evaluation when the ill-conditioned variables are updated on the process. Therefore, this process often falls into local extrema when solving a large-scale system.

In this paper, we intend to enhance the numerical models in a power system with the aid of the resurgent GAs. This paper deals with the optimization of nonlinear algebraic equation. Since it needs a exact solution, the solution procedure will be rather complex and finding solution is more difficult than those of conventional genetic optimizations such as, economic load dispatch and unit commitment which are relatively concerns on the better optimum to the given problem. By this way, we have a defect of consuming much compution time to calculate the errors in each string and binary representation of variables due to the random trasition property of GAs which is not found in mathematical optimization. In order to secure convergence and solution itself in GAs optimization, comparative approach with numerical methods will help to confirm GA's effectiveness and propriety.

To proceed with genetic application to the algebraic nonlinear equations, we have decomposed into two major topics. Frist, for the purpose of ensuring the propriety of this approach, a trigonometric objective function which contains multiple solution within a specified range has examined by comparison with mathematical optimization techniques such as, quasi-Newton or Davidon-Flecher-Powell method. Secondly, the study on a power system application using GAs has carried out with solving the state variables eg. voltage magnitude and phase angle in power flow. In genetic operation, each chromosome's fitness is scaled in many ways to prevent the premature convergence. And, we propose the methodology of prescribing the fitness function with an assumption that all the payoff values in the population pool depend upon the Q function defined by the Gaussian cumulative distribution function.

II. An Optimization Using GAs

1. Genetic Algorithms

GAs manipulate a population of potential solution to an optimization or search problem. Specifically, they operate on encoded representations of the solutions, equivalent to the genetic material of individuals in nature, and not directly on the solutions themselves. Each solution is associated with a fitness value that reflects how good it is and compared with other solutions in the population. The higher the fitness value of an individual, the higher the chances of survival and reproduction and the larger

its representation in the subsequent epoch. Recombination of genetic material in GAs is stimulated through a crossover mechanism that exchanges portions between strings. Another operation, called mutation, causes sporadic and random alteration of the bits of strings. Mutation also has a direct analogy from nature and plays the role of regenerating lost genetic materials.

Fundamental to the GAs structure is the encoding mechanism for representing the optimization problem's variables. The encoding mechanism depends upon the nature of the problem variables. For example, when solving for the real-valued problem, the variables assume continuous values, while the variables in a integer-valued problem are binary quantities. In each case, the encoding mechanism should map each solution to a unique binary string.

The objective function, the function to be optimized, provides the information for evaluating each string. However, its range of variables varies from problem to problem. To maintain uniformity over various problem domains, we use the fitness function to normalize the objective function to a convenient range of 0.0 to 1.0. The normalized value of objective function is the fitness of the string, which the reproduction mechanism uses to evaluate the strings of the population.

Using genetic operators which are reproduction, crossover and mutation, the algorithm creates the subsequent epoch from the strings of the current population. As a process exit criterion, a predefined number of epoch is generally used and then, the results of the fittest chromosome through the entire epoch are designated as an optimum. Other criteria, such as the difference between the maximum and minimum fitness, the rate of increase in maximum fitness and the ratio between the average and maximum fitness could be used.

Fig. 1 summarizes the working of GAs in pseudo-C code, which has the following components: a population of binary strings, genetic control parameters, a fitness function, genetic operators and a mechanism to encode the solutions as binary strings.

```
void main(void)
{
    int    epoch;

    initialize(old_population);
    evaluate(old_population);
    for (epoch=1;  epoch  <  MAX_EPOCH;  epoch++)
    {
        new_population = reproduction(old_population);
        crossover(new_population);
        mutation(new_population);
        evaluate(new_population);
        old_population = new_population;
    }
}
```

Fig. 1. The working of GAs represented in pseudo-C code.

2. A Fitness Mapping with the Q Function

There can be many ways how to determine the fitness value, since it takes a propensity for problem-dependent characteristics. In minimization problem, an exponential fitness mapping is possible to represent the payoffs. And, just taking simply inversion of the payoffs can be another fitness mapping method. However, it often fails to apply for the problems with multivariable. In this section, we propose the methodology of prescribing the fitness function with an assumption that all the payoff values in the population pool depend upon the error function defined from Gaussian cumulative distribution function(cdf).

$\Phi(x)$ is the cdf of a Gaussian random variable with $m=0$ and $\sigma=1$:

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt . \qquad (1)$$

Therefore, any probability involving an arbitrary Gaussian random variable can be expressed in terms of $\Phi(x)$. Q function is defined by

$$Q(x) \doteq 1 - \Phi(x) \qquad (2)$$
$$= \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{-t^2/2} dt .$$

According to the Börjesson's work[7], it has been found to give good accuracy for $Q(x)$ over the range $0 < x < \infty$.

$$Q(x) \simeq \left(\frac{1}{(1-a)x + a\sqrt{x^2+b}} \right) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \qquad (3)$$

where, $a = 1/\pi$ and $b = 2\pi$.

As shown in *Fig. 2* (b), the *region* is the normalized and sorted errors or costs of the objective function and the *domain* is their corresponding fitness values. The chromosome which has minimum errors in the population is assigned to fitness 1.0 and, the chromosome with maximum errors is mapped into fitness 0.0.

3. Approaches to Trigonometric Function

1) A Unimodal Case

For the purpose of taking prior steps to resolve the multiple solutions occurred in power systems, we have made an approach of determining the optimal points in a trigonometric objective function. The objective function to be used for GAs optimization is

$$\text{minimize} \quad \frac{\sqrt{(1-x_1 x_2 \sin x_2)^2 + (1-x_1 x_2 \cos x_1)^2}}{2} \qquad (4)$$

subject to $-1 \le x_1 \le 1, \quad -1 \le x_2 \le 1$

Since the described problem is kind of multi-variable optimization, the whole chromosome can be constructed by concatenating each. And then, each chromosome's representation
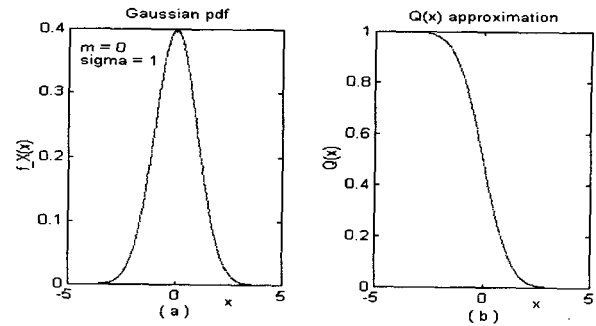

Fig. 2. The Q(x) for a fitness mapping.

of the variable must be converted to real numbers in an appropriate range. Each allele is represented in binary code, so we can get decimal value between 0 and $2l-1$ where l is the chromosome length. And then, to map this decimal value into real one, we can use the below linear function to map linearly from the integer region $[i_{min}, i_{max}]$ to the real domain $[r_{min}, r_{max}]$.

$$r = \frac{r_{max} - r_{min}}{i_{max} - i_{min}} i + r_{min} \qquad (5)$$

where, $i_{min} = 0, \quad i_{max} = 2^l - 1$

For more accurate r precision, the chromosome length will be large enough. However, with a longer chromosome, GAs find difficulty in reaching a near optimum, since a genetic search exploits schemata which represent hyperplanes, and an increase in the size of chromosome increases the amount of space that the algorithm needs to explore to find good schemata. Since the optimal population size is a function of the chromosome length for better schema processing, increasing the chromosome length may incur bigger size of population which leads to inefficiency in view of computation memory and time. Therefore, you should be determine the appropriate chromosome length with regard to the size of population.

Encoding procedure is vice versa. In these examples, the chromosome length is 15 so, the precision of the region is given as 6.1×10^{-5}. The reproduction and crossover method used in this example are remainder stochastic sampling with replacement and two-point crossover, respectively. And, for a fitness scaling the linearization method is used.

Fig. 3 shows the fitness values from the proposed error distribution processing. Since all the fitness values in any epoch can be determined by the error distribution function such as, a, b, c and d, the average fitness value has near 0.5. All errors in epoch 10 has an a-distributed population and fitness mapping is carried out with distribution curve a. In each epoch, fitness mapping from the errors is carried out according to thier distribution curve. In this processing, the convergence can be checked by whether an error of maximally fitted string lies within error tolerance.
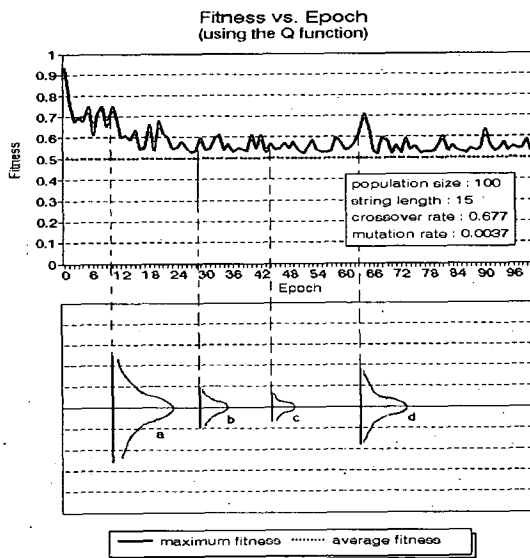
Fig. 3. The fitness value with the Q function.

Table 1. Comparison between numerical methods and proposed GAs

iteration	function values			epoch in GAs
	quasi-Newton	DFP	proposed GAs	
4	1.71442	1.71442	0.03634	4
11	0.68091	0.68091	0.01462	11
16	0.67884	0.67890	0.01632	16
23	0.54227	0.31334	0.00170	23
26	0.26496	0.24227	0.00169	26
31	0.11665	0.20201	0.00535	31
37	0.02017	0.04720	0.00036	37
42	0.01810	0.04092	0.00093	42
50	0.00803	0.03342	0.00125	50
56	0.00690	0.00472	0.00076	56
61	0.00524	0.00147	0.00060	61
67	0.00101	0.02466	0.00052	67
72	0.00067	0.00231	0.00067	72
77	0.00031	0.00027	0.00019	77
82	0.00016	0.00013	0.00010	78
87	0.00009	0.00010		

For a comparison, the quasi-Newton method and DFP method are used. The initial point for the numerical process is given as (-1, 2.5) and, the global minimum is (0.5534, 2.1243). *Table 1* shows the comparative results with proposed GAs. The results imply that GAs can have pretty good outcome over the well-known numerical methods in a specified region.

2) A Multimodal Case

Equation (6) contains many local minima because of the high
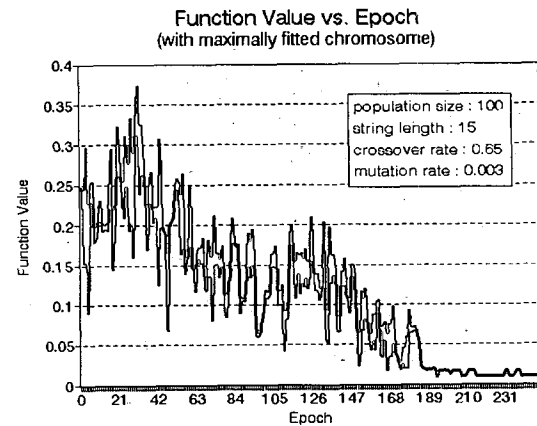

Fig. 4. The function value vs epoch in genetic optimization.

order terms inside the trigonometric function. In this case, well-known calculus-based approaches fail to find the global points and, if ever it does, it would have a very complex and elaborate procedure.

minimize

$$\frac{\sqrt{(1-x_1x_2\sin(500x_2))^2 + (1-x_1x_2\cos(500x_1))^2}}{2} \tag{6}$$

subject to $-1 \leq x_1 \leq 1, \quad -1 \leq x_2 \leq 1$.

Fig. 4 shows that even a multimodal case GAs can find optima with simple but powerful their operations. In the early epochs within 100, GAs find candidates near the optima, after that, converge to one point with most potential candidate. From this experiment, an applicability to the power system optimization which always have trouble with the multiple solutions is proved even if it contains nonlinear algebraic equations which should be resolved with an exact solution. However, to resolve a more complicated problem in a power system, further considerations for the GAs dynamics remain also.

III. The Power Flow Using GAs

In the power flow which is primarily concerned with calculating complex power flowing throughout the transmission networks, the voltage magnitude and phase angle of the slack bus are dependent variables and, voltage magnitude of the generator bus is specified in advance. Therefore, in a n-bus system, the total number of independent variable is $2n-(2s+g)$ where, s is the number of slack bus and, g is the number of generator bus. If we did not consider reactive power constraints in generator bus for brevity of the problem, the objective function can be formulated as follows.

minimize

$$\frac{\sqrt{\sum\limits_{i\neq slack}(\Delta P_i)^2 + \sum\limits_{i=load}(\Delta Q_i)^2}}{N} \quad i = 1, 2, \cdots, n \quad (7)$$

subject to $\Delta P_i = P_i^{specified} - P_{Ti}$

$\Delta Q_i = Q_i^{specified} - Q_{Ti}$

where, $P_{Ti} = f(V, \delta)$

$Q_{Ti} = g(V, \delta)$

N : number of independent variables

In order to use GAs for the power flow problem, the first step is to encode the independent variable to the chromosome with an appropriate length. And then, a fitness function must be prescribed for better selection of the chromosome in reproduction process. The chromosome structure could be organized, as shown in *Fig. 5*. First, every independent variable is represented in one chromosome, and it is operated within one population pool. Secondly, changing the position of the independent variable is another kind of chromosome structure. In a third, taking up two population pool, the V_i (voltage magnitude) chromosome and the δ_i (phase angle) chromosome can be operated separately. Finally, with multi-population pool about the independent variable, each chromosome is operated respectively.

To quantify the effectiveness of GAs and inspect the convergence, the performance measure M is given by

$$M = \frac{\sum\limits_{t=1}^{T} F_{max}^t(V, \delta)}{T} \quad (8)$$

where, $F_{max}^t(V, \delta)$: payoff with max fitness in epoch t

T : predefined number of epoch

In words, this performance is an average evaluation of the objective function up to the current epoch. In this paper, this measure is used to determine the optimal genetic control parameter sets.

IV. The Case Study

For the case study, 5-bus, 14-bus and 25-bus systems are selected. *Fig. 6* summarizes the working of power flow computation using GAs represented in pseudo-C code.

Through the experiments, the states when the population size 300, crossover rate 1.0 or 0.8 and, mutation rate 0.001 or 0.0005 turn out to be proper. The remainder stochastic sampling without replacement where added on the elite population at the rate of 0.01 and two-point crossover occurred on the unit chromosome are used.

Fig. 7 and 8 show all the errors in the population pool where the randomly selected epoch have the Gaussian *pdf* characteristics. These characteristics validate the propriety of the proposed fitness mapping. In *Fig. 7*, the negative side of the normalized
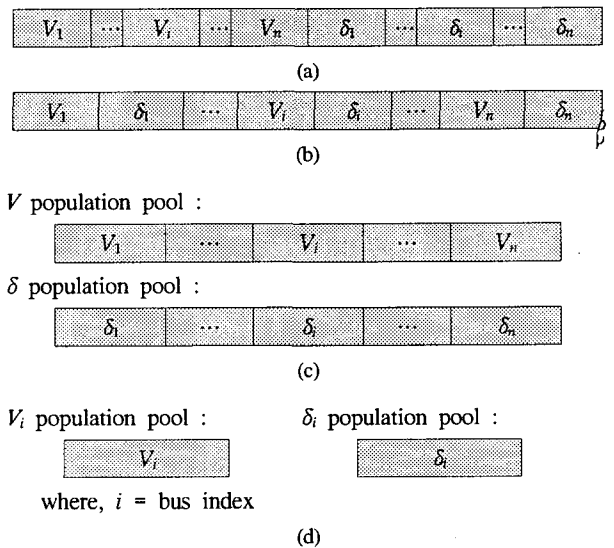

Fig. 5. Chromosome structure for the variable encoding.

```
void main(void)
{
    int     epoch;

    randomize_module(independent_variables);
    for (epoch=1; epoch < MAX_EPOCH; epoch++)    {
        GAs_process_module(objective_function);
        if (acceptable_errors)
            power_flow_module(independent_variables);
    }
}
```
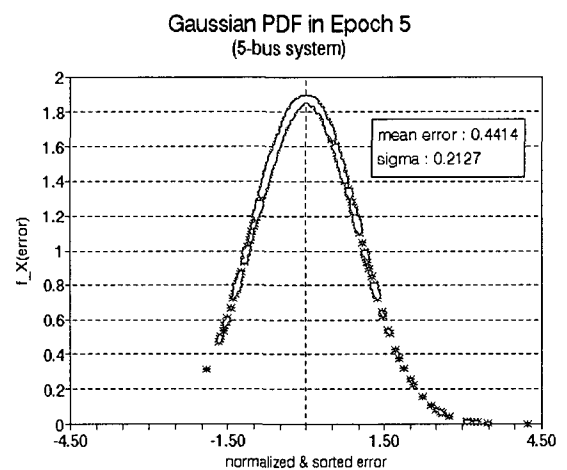
Fig. 6. Power flow procedure in pseudo-C code.

Fig. 7. The 5th epochal population state in 5-bus system.

and sorted errors is to be higher fitness than the average. *Fig. 8* implies that all the chromosomes have an identical payoff value
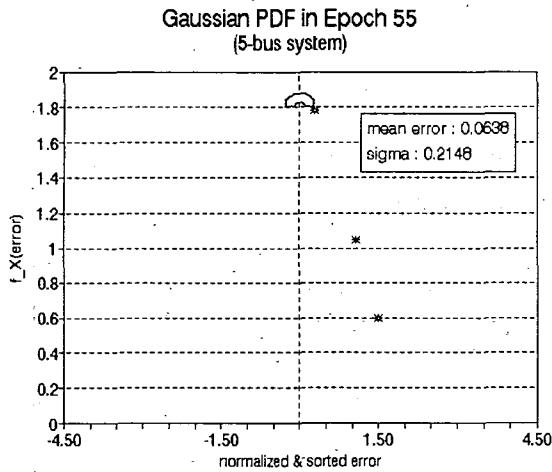
Fig. 8. The 55th epochal population state in 5-bus system.

Table 2. Complex power flowing report using proposed GAs in 14-bus system.

============POWER FLOWS============			==========POWER LOSSES=			
P_tran	Q_tran	[Bus-Bus]	P_tran	Q_tran	P_loss	Q_loss
1.5450	-0.1985	[1 2]	-1.503y	0.2672	0.0417	0.068
0.7867	-0.1803	[1 5]	-0.7558	0.2522	0.0309	0.0719
0.7106	0.0378	[2 3]	-0.6887	0.0082	0.0219	0.0460
0.5593	-0.2372	[2 4]	-0.5401	0.2540	0.0191	0.0168
0.4172	-0.2633	[2 5]	-0.4050	0.2627	0.0122	-0.0006
-0.2530	-0.1865	[3 4]	0.2590	0.1650	0.0061	-0.0214
-0.6378	-0.0574	[4 5]	0.6427	0.0584	0.0049	0.0011
0.2813	-0.0031	[4 7]	-0.2813	0.0179	0.0000	0.0148
0.1594	0.0363	[4 9]	-0.1594	-0.0230	0.0000	0.0133
0.4431	0.0057	[5 6]	-0.4431	0.0379	0.0000	0.0435
0.0738	0.0711	[6 11]	-0.0730	-0.0692	0.0009	0.0018
0.0789	0.0296	[6 12]	-0.0782	-0.0280	0.0008	0.0016
0.1783	0.0904	[6 13]	-0.1760	-0.0859	0.0023	0.0045
0.0000	-0.1926	[7 8]	0.0000	0.1985	0.0000	0.0058
0.2812	0.1752	[7 9]	-0.2812	-0.1644	0.0000	0.0108
0.0524	0.0076	[9 10]	-0.0523	-0.0074	0.0001	0.0002
0.0927	0.0139	[9 14]	-0.0917	-0.0117	0.0010	0.0022
-0.0378	-0.0506	[10 11]	0.0381	0.0513	0.0003	0.0007
0.0172	0.0120	[12 13]	-0.0171	-0.0120	0.0001	0.0001
0.0582	0.0399	[13 14]	-0.0574	-0.0383	0.0008	0.0016
==						

N.B.   Line and bus input data are presented in Appendix.

as the GAs converge. This phenomenon seems to conform to the building blocks hypothesis.

Major concerns on GAs mechanics in this paper can be enumerated, as follows. First, the effects as the modification of the chromosome structure are presented and their major results are analyzed. Secondly, the optimal choice of the genetic control
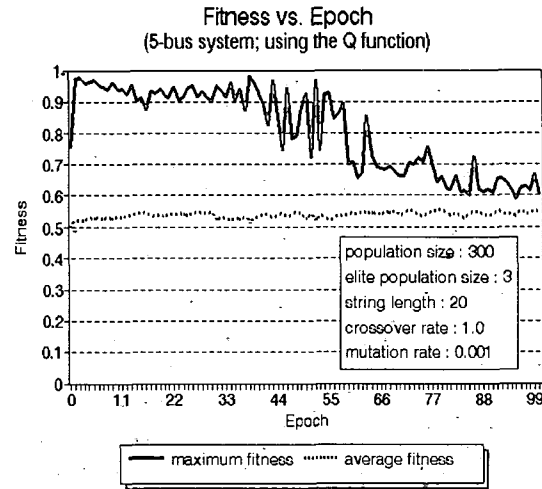

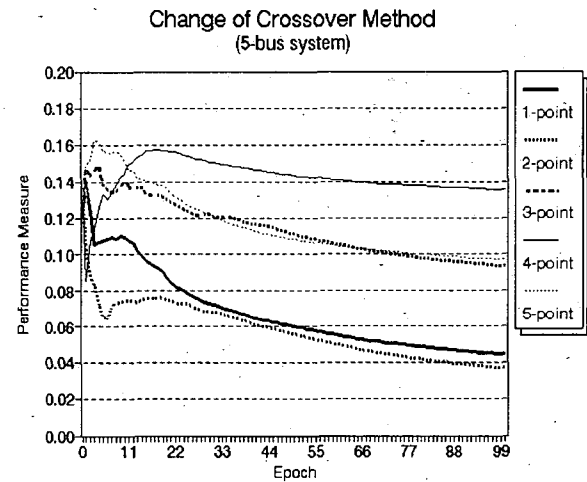**Fig. 9.** The fitness value in 5-bus system.

Fig. 10. On-line performance with changing the crossover method (5-bus).

Complex power flowing of 14-bus system is presented in *Table 2. Table 3* shows the bus states comparison between Newton-Raphson method and GAs in the 25-bus system. You can see that genetic power flow which operates not using the derivative information eg. Jacobian matrix construction but using random genetic processing has the comparable outcomes to that of Newton-Raphson's. From above case studies, the fitness mapping based on error distribution has a validity on GAs processing.

Table 4 shows on-line performance M and computation time with pentium-133MHz CPU in the genetic power flow where population size 300, chromosome length 25, P_c 0.95 and P_m 0.003. The remainder stochastic sampling without replacement where added on the elite population at the rate of 0.01 and two-point crossover occurred on the unit chromosome are used in this

Table 3. Bus states(voltages) comparison between N-R and GAs in the 25-bus system.

| | | [Newton-Raphson Method] | | | | [Proposed GAs] | | | |
| ==BUSES== | | ===Initial==== | | ===Optimized=== | | ===Initial==== | | ===Optimized=== | |
[#]	Type	Mag	Ang	Mag	Ang	Mag	Ang	Mag	Ang
[1]	LOAD	1.0000	0.0000	1.0371	-0.1120	M_RAN	A_RAN	1.0351	-0.1120
[2]	LOAD	1.0000	0.0000	1.0373	-0.0590	M_RAN	A_RAN	1.0373	-0.0590
[3]	LOAD	1.0000	0.0000	1.0099	-0.0463	M_RAN	A_RAN	1.0199	-0.0463
[4]	LOAD	1.0000	0.0000	1.0302	0.0935	M_RAN	A_RAN	1.0302	0.0935
[5]	LOAD	1.0000	0.0000	1.0387	0.0561	M_RAN	A_RAN	1.0387	0.0561
[6]	LOAD	1.0000	0.0000	1.0492	0.0570	M_RAN	A_RAN	1.0492	0.0470
[7]	LOAD	1.0000	0.0000	1.0459	0.0549	M_RAN	A_RAN	1.0412	0.0749
[8]	LOAD	1.0000	0.0000	1.0369	-0.0600	M_RAN	A_RAN	1.0669	-0.0300
[9]	LOAD	1.0000	0.0000	0.9980	-0.0315	M_RAN	A_RAN	1.0030	-0.0415
[10]	LOAD	1.0000	0.0000	1.0339	-0.1228	M_RAN	A_RAN	1.0339	-0.1285
[11]	LOAD	1.0000	0.0000	1.0375	0.0123	M_RAN	A_RAN	1.0375	0.0321
[12]	LOAD	1.0000	0.0000	1.0290	0.0885	M_RAN	A_RAN	1.0291	0.0885
[13]	LOAD	1.0000	0.0000	0.9974	-0.0490	M_RAN	A_RAN	0.9974	-0.0490
[14]	GENE	1.0500	0.0000	1.0500	0.1088	1.0500	A_RAN	1.0500	0.1088
[15]	GENE	1.0500	0.0000	1.0500	0.1472	1.0500	A_RAN	1.0500	0.1472
[16]	GENE	1.0500	0.0000	1.0500	-0.0235	1.0500	A_RAN	1.0500	-0.0235
[17]	GENE	1.0500	0.0000	1.0500	0.1530	1.0500	A_RAN	1.0500	0.1530
[18]	LOAD	1.0000	0.0000	1.0439	0.0759	M_RAN	A_RAN	1.0439	0.0759
[19]	LOAD	1.0000	0.0000	1.0478	0.0766	M_RAN	A_RAN	1.0378	0.0746
[20]	LOAD	1.0000	0.0000	1.0348	-0.0828	M_RAN	A_RAN	1.0148	-0.0828
[21]	LOAD	1.0000	0.0000	1.0306	-0.1015	M_RAN	A_RAN	1.0396	-0.1415
[22]	LOAD	1.0000	0.0000	1.0452	0.0018	M_RAN	A_RAN	1.0438	0.0218
[23]	LOAD	1.0000	0.0000	1.0456	0.0664	M_RAN	A_RAN	1.0254	0.0664
[24]	LOAD	1.0000	0.0000	1.0475	0.0496	M_RAN	A_RAN	1.0472	0.0396
[25]	SLCK	1.0500	0.0000	1.0500	0.0000	1.0500	0.0000	1.0500	0.0000

N.B.   [GAs Parameters]
POP_SIZE = 300, Elite POP_SIZE = 3, CHROM_LEN = 25, CROSSOVER_RATE = 0.950,
MUTATION_RATE = 0.001, M_RAN RANGE : [0.90, 1.10], A_RAN RANGE : [-0.30, 0.30]
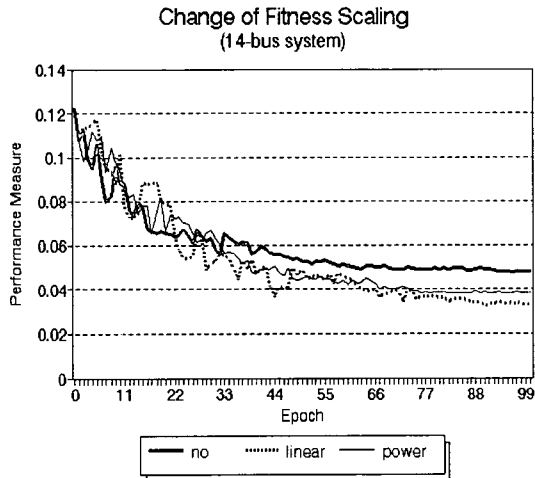Remainder Stochastic Sampling Without Replacement, 2-Point Crossover, Linear Fitness Scaling

Change of Fitness Scaling
(14-bus system)

Fig. 11. On-line performance with fitness scaling (14-bus).

in this experiments. From the results of simulation, genetic power flow which is calculated with the maximally fitted chromosome can be useful when the on-line performance M is less than 0.45.

Major concerns on GAs mechanics in this paper can be enumerated, as follows. First, the effects as the modification of the chromosome structure are presented and their major results

Table 4. On-line performance M & computation time in genetic power flow.

system epoch	5-bus	14-bus	25-bus
100	0.034	0.036	0.064
200	0.028	0.048	0.042
300	0.037	0.053	0.051
[min:sec.mili]	[2:19.49]	[5:01.31]	[7:43.68]

are analyed. Secondly, the optimal choice of the genetic control parameter sets such as population size, chromosome length, crossover rate and, mutation rate in our problem is determined by the contemplated experiments. In a third, the effect of adding elite population is considered. Finally, the various reproduction and crossover mechanisms are compared and analyzed in order to get the best performance from the GAs.

V. Conclusions

The key issues in this paper can be summarized in two ways. First, in the genetic processing, the methodology of prescribing the fitness function using the Q function defined from the Gaussian cdf is presented. Secondly, the proposed method is applied to

calculate the power flow in sample systems and, the results of computational experiments suggest an applicability of GAs to the more complex power system problems.

In genetic power flow, we obtain a comparable result to that of Newton-Raphson but it consumes much computation time in encoding and decoding procedures. The result of case studies suggests us a direct applicability of GAs to complicated numerical problem even if it contains nonlinear algebraic equations as a constraint. Further considerations remain for an exact description of GAs dynamics and brevity in genetic representation which can directly affect on the fast and robust processing of GAs.

Appendix

Table A.1. 14-bus system line data.

Bus - Bus		R	X	$Y_c / 2$	Tap
1	2	0.01938	0.05917	0.02640	1.00000
1	5	0.05403	0.22304	0.02460	1.00000
2	3	0.04699	0.19797	0.02190	1.00000
2	4	0.05811	0.17632	0.01870	1.00000
2	5	0.05695	0.17388	0.01700	1.00000
3	4	0.06701	0.17103	0.01730	1.00000
4	5	0.01335	0.04211	0.00640	1.00000
4	7	0.00000	0.20912	0.00000	0.97800
4	9	0.00000	0.55618	0.00000	0.96900
5	6	0.00000	0.25202	0.00000	0.93200
6	11	0.09498	0.19890	0.00000	1.00000
6	12	0.12291	0.25581	0.00000	1.00000
6	13	0.06615	0.13027	0.00000	1.00000
7	8	0.00000	0.17615	0.00000	1.00000
7	9	0.00000	0.11001	0.00000	1.00000
9	10	0.03181	0.08450	0.00000	1.00000
9	14	0.12711	0.27038	0.00000	1.00000
10	11	0.08205	0.19207	0.00000	1.00000
12	13	0.22092	0.19988	0.00000	1.00000
13	14	0.17093	0.34802	0.00000	1.00000

Table A.2. 14-bus system input bus data for the GAs process.

| Bus#	Bus Type	V_{mag}	V_{ang}	P_{gen}	Q_{gen}	P_{load}	Q_{load}
1	Slack	1.0600	0.0000	2.3240	-0.1690	0.0000	0.0000
2	Generator	1.0450	A_RAN	0.4000	0.4240	0.2170	0.1270
3	Generator	1.0100	A_RAN	0.0000	0.2340	0.9420	0.1900
4	Load	M_RAN	A_RAN	0.0000	0.0000	0.4780	-0.0390
5	Load	M_RAN	A_RAN	0.0000	0.0000	0.0760	0.0160
6	Generator	1.0700	A_RAN	0.0000	0.1220	0.1120	0.0750
7	Load	M_RAN	A_RAN	0.0000	0.0000	0.0000	0.0000
8	Generator	1.0900	A_RAN	0.0000	0.1740	0.0000	0.0000
9	Load	M_RAN	A_RAN	0.0000	0.0000	0.2950	0.1660
10	Load	M_RAN	A_RAN	0.0000	0.0000	0.0900	0.0580
11	Load	M_RAN	A_RAN	0.0000	0.0000	0.0350	0.0180
12	Load	M_RAN	A_RAN	0.0000	0.0000	0.0610	0.0160
13	Load	M_RAN	A_RAN	0.0000	0.0000	0.1350	0.0580
14	Load	M_RAN	A_RAN	0.0000	0.0000	0.1490	0.0500

M_RAN, A_RAN : random number within a specified range

References

[1] D. E. Goldberg, Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley, 1989.

[2] Proc. of Evolutionary Computation, IEEE '94 ICEC, Vol. 1, 2, June 1994.

[3] G. B. Sheble and D. E. Walters, "*Genetic Algorithm Solution of Economic Dispatch with Valve Point Loading*", IEEE Transactions on Power Systems, Vol. 8, No. 3, August 1993.

[4] K. Iba, "*Reactive Power Optimization by Genetic Algorithm*", IEEE Transactions on Power Systems, Vol. 9, No. 2, May 1994.

[5] D. Dasgupa and D. R. McGregor, "*Thermal Unit Commitment Using Genetic Algorithms*", IEE Proc. of Gener. Transm. Distrib., Vol. 141, No. 5, Sep. 1994.

[6] V. Miranda, J. V. Ranito and L. M. Proença, "*Genetic Algorithms in Optimal Multistage Distribution Network Planning*", IEEE Transactions on Power Systems, Vol. 9, No. 4, November 1994.

[7] Xiaodong Yin and Noël Germay, "*Investigations on Solving the Load Flow Problem by Genetic Algorithms*", Electic Power Systems Research, Vol. 22, pp. 151-163, 1991.

[8] P. O. Börjesson and C. E. W. Sundberg, "*Simple Approximations of the Error Function Q(x) for Communication Applications*", IEEE Transactions on Communications, pp. 639-643, Mar. 1979.
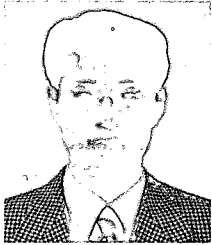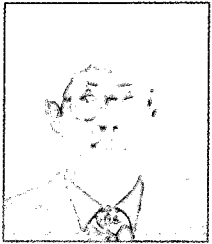
Seungchan Chang received the B.S. and M.S. degree in Electrical Engineering from Hong-Ik University, Seoul, Korea in 1992 and 1995, respectively. He was a researcher of KAITECH in 1995. Since 1996, he has worked for RaCER(R&D Management Center for Energy and Resources). He is currently working toward the Ph.D. degree in Electrical Engineering at Hong-Ik University. His present research interests are power system operation and planning especially in DSM/IRP strategies.
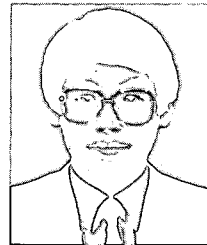
Jung-Hoon Kim received the B.S., M.S. and Ph.D. degree in Electrical Engineering from Seoul National University, Korea in 1978, 1981 and 1985, respectively. He worked as a researcher in KERI from 1978 to 1980. He was a visiting professor at Penn State University, U.S.A, from 1987 to 1989. Since 1981, he has been with the Dept. of Electrical Engineering in Hong-Ik University as a professor. His research interests are the analysis of power system stability, power system planning and energy economics of power facilities.

Jae-Yoon Lim received the B.S. and M.S. degree in Electrical Engineering from Chungbuk National Unversity, Cheongju, Korea in 1984 and 1986, respectively and Ph.D. degree in Hongik University, Seoul, Korea in 1995. Since 1991, he has been with the Dept. of Eletrical Engineering at Chungnam Junior College, Taejeon, Korea as a professor. He is a member of KIEE, IEEJ and IEEE PES. His research interests are load forecasting, load modeling using load composition rate, load identification and application of fuzzy & neural networks.