An Improved Genetic Algorithm for Generation Expansion Planning

Jong-Bae Park, Young-Moon Park, Jong-Ryul Won, and Kwang Y. Lee

***Abstract*—This paper presents a development of an improved genetic algorithm (IGA) and its application to a least-cost generation expansion planning (GEP) problem. Least-cost GEP problem is concerned with a highly constrained nonlinear dynamic optimization problem that can only be fully solved by complete enumeration, a process which is computationally impossible in a real-world GEP problem. In this paper, an improved genetic algorithm incorporating a stochastic crossover technique and an artificial initial population scheme is developed to provide a faster search mechanism. The main advantage of the IGA approach is that the "curse of dimensionality" and a local optimal trap inherent in mathematical programming methods can be simultaneously overcome. The IGA approach is applied to two test systems, one with 15 existing power plants, 5 types of candidate plants and a 14-year planning period, and the other, a practical long-term system with a 24-year planning period.**

***Index Terms*—Generation expansion planning, genetic algorithm, global optimization, improved genetic algorithm.**

I. Introduction

GENERATION expansion planning (GEP) is one of the most important decision-making activities in electric utilities. Least-cost GEP is to determine the minimum-cost capacity addition plan (i.e., the type and number of candidate plants) that meets forecasted demand within a prespecified reliability criterion over a planning horizon.

A least-cost GEP problem is a highly constrained nonlinear discrete dynamic optimization problem that can only be fully solved by complete enumeration in its nature [1]–[3]. Therefore, every possible combination of candidate options over a planning horizon must be examined to get the optimal plan, which leads to the computational explosion in a real-world GEP problem.

To solve this complicated problem, a number of salient methods have been successfully applied during the past decades. Masse and Gilbrat [4] applied a linear programming approach that necessitates the linear approximation of an objective function and constraints. Bloom [5] applied a mathematical programming technique using a decomposition method, and solved it in a continuous space. Park *et al.* [6] applied the Pontryagin's maximum principle whose solution also lies in a continuous space. Although the above-mentioned mathematical programming methods have their own advantages, they possess one or both of the following drawbacks in solving a GEP problem. That is, they treat decision variables in a continuous space. And there is no guarantee to get the global optimum since the problem is not mathematically convex. Dynamic programming (DP) based framework is one of the most widely used algorithms in GEP [1]–[3], [7], [8]. However, so-called "the curse of dimensionality" has interrupted direct application of the conventional full DP in practical GEP problems. For this reason, WASP [1] and EGEAS [2] use a heuristic tunneling technique in the DP optimization routine where users prespecify states and successively modify tunnels to arrive at a local optimum. David and Zhao developed a heuristic-based DP [7] and applied the fuzzy set theory [8] to reduce the number of states. Recently, Fukuyama and Chiang [9] and Park *et al.* [10] applied genetic algorithm (GA) to solve sample GEP problems, and showed promising results. However, an efficient method for a practical GEP problem that can overcome a local optimal trap and the dimensionality problem simultaneously has not been developed yet.

GA is a search algorithm based on the hypothesis of natural selections and natural genetics [11]. Recently, a global optimization technique using GA has been successfully applied to various areas of power system such as economic dispatch [12], [13], unit commitment [14], [15], reactive power planning [16]–[18], and power plant control [19], [20]. GA-based approaches for least-cost GEP have several advantages. Naturally, they can not only treat the discrete variables but also overcome the dimensionality problem. In addition, they have the capability to search for the global optimum or quasioptimums within a reasonable computation time. However, there exist some structural problems in the conventional GA, such as premature convergence and duplications among strings in a population as generation progresses [11].

In this paper, an improved genetic algorithm (IGA), which can overcome the aforementioned problems of the conventional GA to some extents, is developed. The proposed IGA incorporates the following two main features. First, an artificial creation scheme for an initial population is devised, which also takes the random creation scheme of the conventional GA into account. Second, a stochastic crossover strategy is developed. In this scheme, one of the three different crossover methods is randomly selected from a biased roulette wheel where the weight of each crossover method is determined through pre-performed experiments. The stochastic crossover scheme is similar to the stochastic selection of reproduction candidates from a mating pool.

Manuscript received March 3, 1999; revised July 19, 1999. This work was supported in part by the Korea Science and Engineering Foundation and the National Science Foundation under Grants INT-9605028 and ECS-9705105.

J.-B. Park is with the Department of Electrical Engineering, Anyang University, Anyang-City, 708-113, Korea.

Y.-M. Park is with the School of Electrical Engineering, Seoul National University, Seoul, 151-742, Korea.

J.-R. Won is with the Korea Electric Power Research Institute, Korea Electric Power Corporation, Taejon, 305-380, Korea.

K. Y. Lee is with the Department of Electrical Engineering, The Pennsylvania State University, University Park, PA 16802 USA.

Publisher Item Identifier S 0885-8950(00)07722-1.

The results of the IGA are compared with those of the conventional simple genetic algorithm, the full DP, and the tunnel-constrained DP employed in WASP.

II. Formulation of the Least-Cost GEP Problem

Mathematically, solving a least-cost GEP problem is equivalent to finding a set of optimal decision vectors over a planning horizon that minimizes an objective function under several constraints. The GEP problem to be considered is formulated as follows [6]:

$$\underset{U_1,\cdots,U_T}{\text{Min}} \sum_{t=1}^{T} \left\{ f_t^1(U_t) + (f_t^2)(X_t) - f_T^3(U_T) \right\} \tag{1}$$

$$\text{s.t.} \quad X_t = X_{t-1} + U_t \quad (t = 1, \cdots, T) \tag{2}$$

$$\text{LOLP}(X_t) < \varepsilon \quad (t = 1, \cdots, T) \tag{3}$$

$$\underline{R} \le R(X_t) \le \overline{R} \quad (t = 1, \cdots, T) \tag{4}$$

$$\underline{M_t^j} \le \sum_{i \in \Omega_j} x_t^i \le \overline{M_t^j} \quad (t = 1, \cdots, T \text{ and } j = 1, \cdots, J) \tag{5}$$

$$0 \le U_t \le \overline{U_t} \quad (t = 1, \cdots, T) \tag{6}$$

where

- T: number of periods (years) in a planning horizon,
- J: number of fuel types,
- Ω_j: index set for jth fuel type plant,
- X_t: cumulative capacity [MW] vector of plant types in year t,
- x_t^j: cumulative capacity [MW] of ith plant type in year t,
- U_t: capacity addition [MW] vector by plant types in year t,
- $\overline{U_t}$: maximum construction capacity [MW] vector by plant types in year t,
- u_t^j: capacity addition [MW] of ith plant in year t,
- $\text{LOLP}(X_t)$: loss of load probability (LOLP) with X_t, in year t,
- $R(X_t)$: reserve margin with X_t, in year t,
- ε: reliability criterion expressed in LOLP,
- $\overline{R}$, $\underline{R}$: upper and lower bounds of reserve margin,
- $\overline{M_t^j}$, $\underline{M_t^j}$: upper and lower bounds of jth fuel type in year t,
- $f_t^1(U_t)$: discounted construction costs [\$] associated with capacity addition U_t in year t,
- $f_t^2(X_t)$: discounted fuel and O&M costs [\$] associated with capacity X_t, in year t,
- $f_T^3(U_t)$: discounted salvage value [\$] associated with capacity addition U_t in year t.

The objective function is the sum of tripartite discounted costs over a planning horizon. It is composed of discounted investment costs, expected fuel and O&M costs and salvage value. To consider investments with longer lifetimes than a planning horizon, the linear depreciation option is utilized [1]. In this paper, five types of constraints are considered. Equation (2) implies state equation for dynamic planning problem [6]. Equations (3) and (4) are related with the LOLP reliability criteria and the reserve margin bands, respectively. The capacity mixes by fuel types are considered in (5). Plant types give another physical constraint in (6), which reflects the yearly construction capabilities.

Although the state vector, X_t, and the decision vector, U_t, have dimensions of MW, we can easily convert those into vectors which have information on the number of units in each plant type. This mapping strategy is very useful for GA implementation of a GEP problem such as encoding and treatment of inequality (6), and illustrated in the following (1) equations:

$$X_t = (x_t^1, \cdots, x_t^N)^T \to X_t' = (x_t'^1, \cdots, x_t'^N)^T \tag{7}$$

$$U_t = (u_t^1, \cdots, u_t^N)^T \to U_t' = (u_t'^1, \cdots, u_t^N)^T \tag{8}$$

where

- N: number of plant types including both existing and candidate plants,
- X_t': cumulative number of units by plant types in year t,
- U_t': addition number of units by plant types in year t,
- $x_t'^i$: ith plant type's cumulative number of units in year t,
- $u_t'^i$: ith plant type's addition number of units in year t.

III. Improved GA for the Least-Cost GEP

A. Overview of Genetic Algorithm

Basically, genetic algorithm is a search mechanism based on the hypothesis of natural selection [11]. GA is an artificial optimization scheme that emulates the hypothetical adaptive nature of natural genetics. GA provides solutions by generating a set of chromosomes referred to as a generation. Each string (chromosome) has its own fitness measure that reflects how well a creature can survive under surrounding environments. The new generation of strings is provided through three major genetic operations—reproduction, crossover and mutation, which provide a powerful global search mechanism. Reproduction is a process in which individual strings are copied into a mating pool according to their fitness values. Crossover, the most important genetic operator, is a structured recombination operation. In the classical one-point crossover, a random position in a string is chosen and all characters to the right of this position are swapped. Mutation, the secondary operator in GA, is an occasional random alteration of the value of a string position. Variations of the simple GA for power system applications can be found in the References [9], [10], [12]–[20]. The improvements on the conventional GA will be described in the subsequent sections.

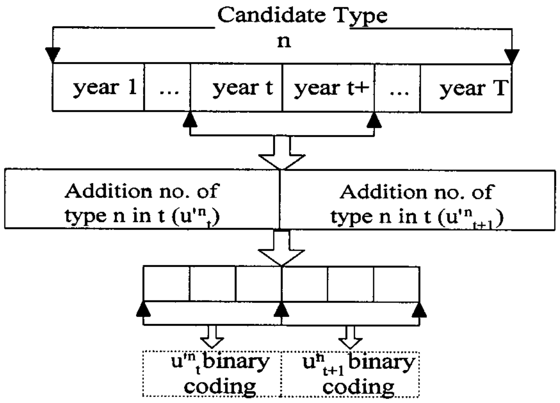

Fig. 1. A substring structure.

B. String Structure

Since it is convenient to use integer values for GA implementation of a GEP problem, the reordered structure of (8) by plant types covering a planning horizon is used for encoding of a string as shown in (9). Here, each element of a string [i.e., $\hat{U}'^n = (u_1'^n, u_2'^n, \cdot, u_T'^n)$ for $n = 1, \cdots, N$] corresponds to a substring and its structure is depicted in Fig. 1.

$$\begin{aligned}\hat{U}' &= \Big(u_1'^1, u_2'^1, \cdot, u_T'^1, \cdots, u_1'^n, u_2'^n, \cdot, u_T'^n, \cdots, \\ &\qquad u_1'^N, u_2'^N, \cdot, u_T'^N\Big)^T \\ &= \Big(\hat{U}'^1, \cdots, \hat{U}'^n, \cdots, \hat{U}'^N\Big)^T. \end{aligned} \tag{9}$$

C. Fitness Function

The objective function or cost of a candidate plan is calculated through the probabilistic production costing and the direct investment costs calculation [1], [2]. The fitness value of a string can be evaluated using the following equation [11], [19]:

$$f = \frac{\alpha}{1 + J} \tag{10}$$

where α is constant and J is objective function of (1).

However, this simple mapping occasionally brings about a premature convergence and duplications among strings in a population, since strings with higher fitness values dominate the occupation of a roulette wheel.

To ameliorate these problems, the following modified fitness function, which normalizes the fitness value of strings into real numbers within [0, 1], is used in this paper [11].

$$f'(i) = \frac{f(i) - f_{\min}}{f_{\max} - f_{\min}} \tag{11}$$

where

$f(i)$	fitness value of string i using (10),
$f_{\max}, f_{\min}$	maximum and minimum fitness value in a generation,
$f'(i)$	modified fitness value of string i.

D. Creation of an Artificial Initial Population

It is important to create an initial population of strings spread out throughout the whole solution space, especially in a large-scale problem. One alternative method could be to increase the population size, which yields a high computational burden. This paper suggests a new artificial initial population (AIP) scheme, which also takes the random creation scheme of the conventional GA into account. The procedures are illustrated in the following and in Table I:

Step 1. Generate all possible binary seeds of each plant type considering (6). For example, if ith plant type has an upper limit of 3 units per year, then generate 4 possible binary seeds (i.e., 00,10, 10, 11).

Step 2. Find the least common multiple (LCM) m from the numbers of the binary seeds of all types, and fill m binary seeds in a look-up table for all plant types and planning years. For example, if three plant types have upper limits of 3, 3 and 5 units per year, respectively, then the numbers of binary seeds are 4, 4, and 6, and m becomes 12.

Step 3. Select an integer within [1, m] at random for each element $u_t'^n$ of a string in (9). Fill the string with the corresponding binary digits, and delete it from the look-up table. Repeat until m different strings are generated.

Step 4. Check the constraints of (3)–(5). If a string satisfies these constraints for all years, then it becomes a member of an initial population. Otherwise, the only parts of the string that violate the constraints in year t are generated at random until they satisfy the constraints. Go to step 3 n times for $n \cdot m$ less than P, where P is the number of strings in a population and n is an arbitrary positive integer.

Step 5. The remaining P–$n{\cdot}m$ strings are created using uniform random variables with binary number $\{0, 1\}$. Go to step 4 to check the constraints and regenerate them if necessary. This process is repeated until all strings, which satisfy the constraints, are generated.

This AIP is based on both artificial and random selection schemes, which allows all possible string structures can be included in an initial population.

E. Stochastic Crossover, Elitism, and Mutation

Most of GA works are based on the Goldberg's simple genetic algorithm (SGA) framework [11]. This paper proposes two different schemes for genetic operation: a stochastic crossover technique and the application of elitism. The stochastic crossover scheme covers three different crossover methods; 1-point crossover, 2-point crossover, and 1-point substring crossover as illustrated in Fig. 2. Each crossover method has its own merits. The 1-point substring crossover can provide diverse bit structures to search solution space, however it easily destroys the string structure that may have partial information on the optimal structure.

Although the 1- and 2-point crossovers can not explore solution space as widely as the above crossover, the probability of destroying an already-found partial optimal structure is very

TABLE I
EXAMPLE OF LOOK-UP TABLE WITH 3 PLANT TYPES FOR 3 PLANNING YEARS

	Type 1 (Upper Limit: 3 Units/Year)			Type 2 (Upper Limit: 3 Units/Year)			Type 3 (Upper Limit: 5 Units/Year)		
m	Year 1	Year 2	Year 3	Year 1	Year 2	Year 3	Year 1	Year 2	Year 3
1	00	00	00	00	00	00	000	000	000
2	01	01	01	01	01	01	001	001	001
3	10	10	10	10	10	10	010	010	010
4	11	11	11	11	11	11	011	011	011
5	00	00	00	00	00	00	100	100	100
6	01	01	01	01	01	01	101	101	101
7	10	10	10	10	10	10	000	000	000
8	11	11	11	11	11	11	001	001	001
9	00	00	00	00	00	00	010	010	010
10	01	01	01	01	01	01	011	011	011
11	10	10	10	10	10	10	100	100	100
12	11	11	11	11	11	11	101	101	101
Generated String 1 : 011100101011000101010									

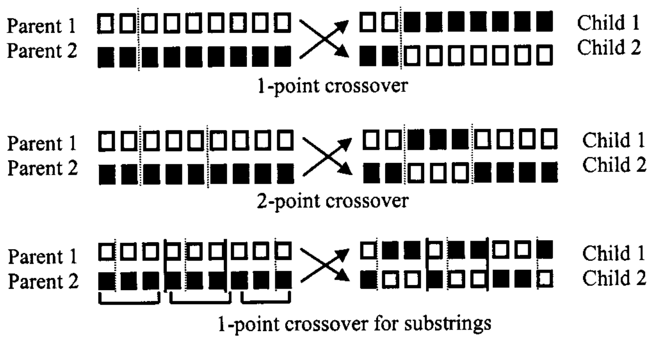

Fig. 2. Three different crossover methods used.

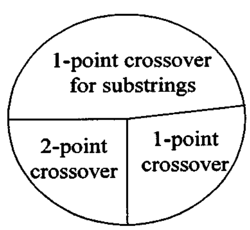

Fig. 3. Roulette wheel for stochastic selection of crossover method.

low. The stochastic crossover strategy is similar to the process of stochastic selection of reproduction candidates from a mating pool. That is, one of the three different crossover methods is selected from a biased roulette wheel, where each crossover method has a roulette wheel slot sized according to its performance. The weight for each crossover method has been determined.

The second feature lies in the application of elitism [15]. The roulette wheel selection scheme gives a reproduction opportunity to a set of recessive members and might not give the set of dominant strings (i.e., an elite group) a chance to reproduce. Furthermore, the application of genetic operations changes the string structures of the fittest solutions. Thus, the best solutions in the current generation might not appear in the next generation. To circumvent these problems an elite group is directly copied into the next generation.

We have applied the conventional mutation scheme [11] where it performs bit-by-bit for the strings undergone the stochastic crossover operator. However, the mutation procedure is not applied to the set of dominant strings to preserve the elitism.

After genetic operations, we check all strings whether they satisfy the constraints of (3)–(5) or not. If any string that violates the constraints of (3)–(5), only the parts of the string that violate the constraints in year t are generated at random until they satisfy the constraints as described in the AIP scheme.

IV. CASE STUDIES

The IGA, SGA, tunnel-constrained dynamic programming (TCDP) employed in WASP, and full dynamic programming (DP) was implemented using the FORTRAN77 language on an IBM PC/Pentium (166NMz) computer.

A. Test Systems Description

The IGA, SGA, TCDP and DP methods have been applied in two test systems: Case I for a power system with 15 existing power plants, 5 types of candidate options and a 14-year study period, and Case 2 for a real-scale system with a 24-year study period. The planning horizons of 14 and 24 years are divided into 7 and 12 stages (two-year intervals), respectively. The forecasted peak demand over the study period is given in Table II.

Tables III and IV show the technical and economic data of the existing plants and candidate plant types for future additions, respectively.

B. Parameters for GEP and IGA

There are several parameters to be pre-determined, which are related to the GEP problem and GA-based programs. In this paper, we use 8.5% as a discount rate, 0.01 as LOLP criteria, and 15% and 60% as the lower and upper bounds for reserve margin, respectively. The considered lower and upper bounds of capacity mix are 0% and 30% for oil-fired power plants, 0% and 40% for LNG-fired, 20% and 60% for coal-fired, and 30% and 60% for nuclear, respectively.

TABLE II
FORECASTED PEAK DEMAND

Stage (Year)	0 (1996)	1 (1998)	2 (2000)	3 (2002)	4 (2004)	5 (2006)	6 (2008)
Peak (MW)	5000	7000	9000	10000	12000	13000	14000
Stage (Year)	-	7 (2010)	8 (2012)	9 (2014)	10 (2016)	11 (2018)	12 (2020)
Peak (MW)	-	15000	17000	18000	20000	22000	24000

TABLE III
TECHNICAL AND ECONOMIC DATA OF EXISTING PLANTS

Name (Fuel Type)	No. of Units	Unit Capacity (MW)	FOR (%)	Operating Cost ($/kWh)	Fixed O&M Cost ($/kW-Mon)
Oil #1 (Heavy Oil)	1	200	7.0	0.024	2.25
Oil #2 (Heavy Oil)	1	200	6.8	0.027	2.25
Oil #3 (Heavy Oil)	1	150	6.0	0.030	2.13
LNG G/T #1 (LNG)	3	50	3.0	0.043	4.52
LNG C/C #1 (LNG)	1	400	10.0	0.038	1.63
LNG C/C #2 (LNG)	1	400	10.0	0.040	1.63
LNG C/C #3 (LNG)	1	450	11.0	0.035	2.00
Coal #1 (Anthracite)	2	250	15.0	0.023	6.65
Coal #2 (Bituminous)	1	500	9.0	0.019	2.81
Coal #3 (Bituminous)	1	500	8.5	0.015	2.81
Nuclear #1 (PWR)	1	1,000	9.0	0.005	4.94
Nuclear #2 (PWR)	1	1,000	8.8	0.005	4.63

TABLE IV
TECHNICAL AND ECONOMIC DATA OF CANDIDATE PLANTS

Candidate Type	Construction Upper Limit	Capacity (MW)	FOR (%)	Operating Cost ($/kWh)	Fixed O&M Cost	Capital Cost ($/kW)	Life Time (yrs)
Oil	5	200	7.0	0.021	2.20	812.5	25
LNG C/C	4	450	10.0	0.035	0.90	500.0	20
Coal (Bitum.)	3	500	9.5	0.014	2.75	1062.5	25
Nuc. (PWR)	3	1,000	9.0	0.004	4.60	1625.0	25
Nuc.(PHWR)	3	700	7.0	0.003	5.50	1750.0	25

TABLE V
PARAMETERS FOR IGA IMPLEMENTATION

Parameters	Value
• Population Size	300
• Maximum Generation	300
• Probabilities of Crossover and Mutation	0.6, 0.01
• Number of Elite Strings	3 (1%)
• Weights of 1-point, 2-point, and 1-point Crossover for Substrings in a Biased Roulette Wheel	0.15:0.15:0.70

Parameters for IGA are selected through experiments. Especially, the dominant parameters such as crossover probabilities and weights for crossover techniques are determined empirically from a test system with a 6-year planning horizon with other data being the same as Cases 1 and 2.

TABLE VI
RESULTS OBTAINED BY EACH CROSSOVER METHOD

Crossover Method	Objective Function in Million Dollars (Errors against Optimal Solution, %) PC = 0.6	PC = 0.7	PC = 0.8
One-point Crossover	5035.53 (0.59%)	5013.50 (0.15%)	5057.30 (1.02%)
Two-point Crossover	5034.89 (0.57%)	5032.98 (0.54%)	5034.89 (0.57%)
One-point Substring Crossover	5012.53 (0.13%)	5012.46 (0.13%)	5010.63 (0.09%)
DP	5006.19		

TABLE VII
RESULTS OBTAINED BY STOCHASTIC CROSSOVER METHOD

Weights	Objective Function in Million Dollars (Errors against Optimal Solution, %) PC = 0.6	PC = 0.7	PC = 0.8
0.05 : 0.05 : 0.90	5007.40* (0.02%)	5010.63 (0.09%)	5007.40 (0.02%)
0.10 : 0.10 : 0.80	**5006.19 (0.00%)**	5010.63 (0.09%)	5012.37 (0.12%)
0.15 : 0.15 : 0.70	5007.40 (0.02%)	**5006.19 (0.00%)**	**5006.19 (0.00%)**
0.20 : 0.20 : 0.60	**5006.19 (0.00%)**	**5006.19 (0.00%)**	5011.79 (0.11%)
0.25 : 0.25 : 0.50	**5006.19 (0.00%)**	5007.40 (0.02%)	5018.37 (0.24%)
0.30 : 0.30 : 0.40	**5006.19 (0.00%)**	5012.46 (0.13%)	5007.40 (0.02%)

* The solution with objective function as 5007.40 million dollars is the second best solution found by dynamic programming.

To decide the weight of each crossover method in a biased roulette wheel for stochastic crossover, nine experiments are performed by changing the probability of crossover from 0.6–0.8, and the results are compared with the optimal solution obtained by the full DP as shown in Table VI.

Among the three crossover methods, the 1-point substring crossover showed the best performance in every case. Thus, we set the 1-point substring crossover with the biggest weight, and others with an equal smaller weight. To determine the weight of each crossover method in a biased roulette wheel, 18 simulations were performed with different weights and crossover probabilities as shown in Table VII.

Among 18 simulations, we have found the optimal solution 7 times and the second best solution 4 times. Furthermore, the optimal or the second best solution is found by applying the stochastic crossover technique when the probability of crossover is 0.6. Also, when the weight of 1-point substring crossover is 0.7 and weights for others are 0.15, it always found optimal or the second best optimal solution. Therefore, we have set the weights in the stochastic crossover technique as 0.15 : 0.15 : 0.70 among the three crossover methods. This choice has resulted in the robustness of the stochastic crossover method.

C. Numerical Results

The developed IGA was applied to two test systems, and compared with the results of DP, TCDP and SGA. Throughout the tests, the solution of the conventional DP is regarded as the global optimum and that of TCDP as a local optimum. Both the

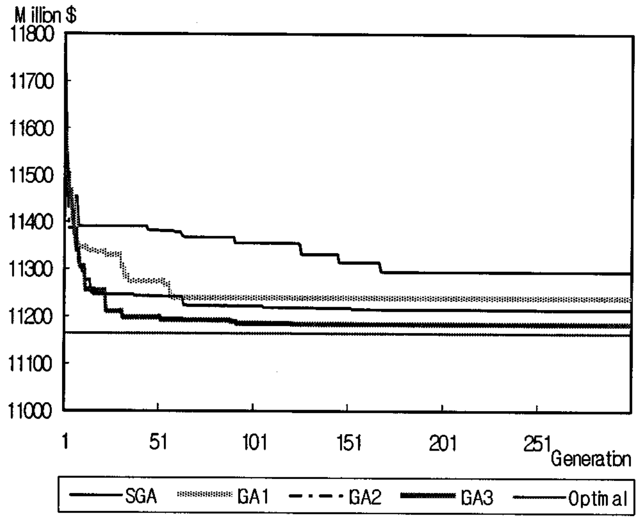

Fig. 4. Convergence characteristics of IGA method in Case 1 system.

TABLE VIII
SUMMARY OF THE BEST RESULTS OBTAINED BY EACH SOLUTION METHOD

Solution Method		Cumulative Discounted Cost (10^6 \$) Case 1 (14-year Study Period)	Case 2 (24-year Study Period)
DP		11164.2	unknown
TCDP		11207.7	16746.7
SGA		11310.5	16765.9
IGA	IGA1	11238.3	16759.2
	IGA2	11214.1	16739.2
	IGA3	11184.2	16644.7

global and a local solution can be obtained in Case 1; however, only a local solution can be obtained by using TCDP in Case 2 since the "curse of dimensionality" prevents the use of the conventional DP.

Fig. 4 illustrates the convergence characteristics of various GA-based methods in Case 1. It also shows the improvement of IGA over SGA. The IGA employing the stochastic crossover scheme (IGA2) has shown better performance than the IGA using the artificial initial population scheme (IGA1). By considering both schemes simultaneously (IGA3), the performance is significantly enhanced.

Table VIII summarizes costs of the best solution obtained by each solution method. In Case 1, the solution obtained by IGA3 is within 0.18% of the global solution costs while the solutions by SGA and TCDP are within 1.3% and 0.4%, respectively. In Case 1 and Case 2, IGA3 has achieved a 0.21% and 0.61% improvement of costs over TCDP, respectively. Although SGA and IGA's have failed in finding the global solution, all IGA's have provided better solution than SGA. Furthermore, solutions of IGA3 are better than that of TCDP in both cases, which implies that it can overcome a local optimal trap in a practical long-term GEP. Table IX summarizes generation expansion plans of Case 1 and Case 2 obtained by IGA3.

The execution time of GA-based methods is much longer than that of TCDP. That is, IGA3 requires approximately 3.7 and 6 times of execution time in Case 1 and Case 2, respectively.

TABLE IX
CUMULATIVE NUMBER OF NEWLY INTRODUCED PLANTS IN CASE 1 AND CASE 2 BY IGA3

Type / Year	Oil (200MW)	LNG C/C (450MW)	Coal (500MW)	PWR (1000MW)	PHWR (700MW)
1998	3 (5)[1]	2 (1)	2 (3)	0 (1)	2 (0)
2000	5 (6)	3 (1)	5 (6)	0 (1)	4 (1)
2002	5 (7)	3 (1)	5 (6)	0 (2)	4 (1)
2004	8 (10)	7 (3)	6 (7)	0 (2)	4 (1)
2006	10 (12)	10 (3)	6 (7)	0 (2)	6 (2)
2008	10 (13)	10 (3)	6 (9)	0 (2)	6 (2)
2010	10 (13)	10 (3)	6 (9)	0 (2)	6 (4)
2012	14	11	8	1	7
2014	17	14	8	1	7
2016	19	15	10	1	9
2018	19	17	10	3	9
2020	20	18	12	3	9

1. The figures within parenthesis denote the results of IGA3 in Case 1.

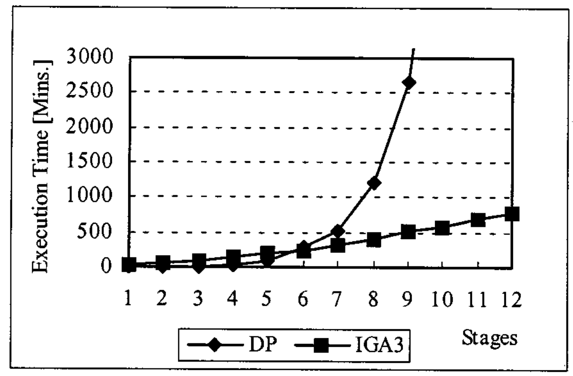

Fig. 5. Observed execution time for the number of stages.

However, it is much shorter than the conventional DP. Fig. 5 shows the observed execution time of IGA3 and DP as the stages are expanded. Execution time of IGA3 is almost linearly proportional to the number of stages while that of DP exponentially increases. In the system with 11 stages, it takes over 9 days for DP, and requires about 1.2 millions of array memories to obtain the optimal solution while it takes only 11 hours by IGA3 to get the near optimum.

The proposed method definitely provides quasioptimums in a long-term GEP within a reasonable computation time. Also, the results of the proposed IGA method are better than those of TCDP employed in the WASP, which is viewed as a very powerful and computationally feasible model for a practical long-term GEP problem. Since a long-range GEP problem deals with a large amount of investment, a slight improvement by the proposed IGA method can result in substantial cost savings for electric utilities.

V. CONCLUSIONS

This paper developed an improved genetic algorithm [12] (IGA) for a long-term least-cost generation expansion planning (GEP) problem. The proposed IGA includes several improvements such as the incorporation of an artificial initial population scheme, a stochastic crossover technique, elitism and scaled fitness function.

The IGA has been successfully applied to long-term GEP problems. It provided better solutions than the conventional SGA. Moreover, by incorporating all the improvements (IGA3), it was found to be robust in providing quasioptimums within a reasonable computation time and yield better solutions compared to the TCDP employed in WASP. Contrary to the DP, computation time of the proposed IGA is linearly proportional to the number of stages.

The developed IGA method can simultaneously overcome the "curse of dimensionality" and a local optimum trap inherent in GEP problems. Therefore, the proposed IGA approach can be used as a practical planning tool for a real-system scale long-term generation expansion planning.

REFERENCES

[1] S. T. Jenkins and D. S. Joy, "Wien automatic system planning package (WASP)—An electric utility optimal generation expansion planning computer code," Oak Ridge National Laboratory, Oak Ridge, TN, ORNL-4945, 1974.

[2] Electric Power Research Institute (EPRI), "Electric generation expansion analysis system (EGEAS),", Palo Alto, CA, EPRI EL-2561, 1982.

[3] S. Nakamura, "A review of electric production simulation and capacity expansion planning programs," *Energy Research*, vol. 8, pp. 231–240, 1984.

[4] P. Masse and R. Gilbrat, "Application of linear programming to investments in the electric power industry," *Management Science*, vol. 3, no. 2, pp. 149–166, 1957.

[5] J. A. Bloom, "Long-range generation planning using decomposition and probabilistic simulation," *IEEE Trans. on PAS*, vol. 101, no. 4, pp. 797–802, 1982.

[6] Y. M. Park, K. Y. Lee, and L. T. O. Youn, "New analytical approach for long-term generation expansion planning based maximum principle and Gaussian distribution function," *IEEE Trans. on PAS*, vol. 104, pp. 390–397, 1985.

[7] A. K. David and R. Zhao, "Integrating expert systems with dynamic programming in generation expansion planning," *IEEE Trans. on PWRS*, vol. 4, no. 3, pp. 1095–1101, 1989.

[8] ——, "An expert system with fuzzy sets for optimal planning," *IEEE Trans. on PWRS*, vol. 6, no. 1, pp. 59–65, 1991.

[9] Y. Fukuyama and H. Chiang, "A parallel genetic algorithm for generation expansion planning," *IEEE Trans. on PWRS*, vol. 11, no. 2, pp. 955–961, 1996.

[10] Y. M. Park, J. B. Park, and J. R. Won, "A genetic algorithms approach for generation expansion planning optimization," in *Proc. of the IFAC Symposium on Power Systems and Power Plant Control*, Pergamon, UK, 1996, pp. 257–262.

[11] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, MA: Addison-Wesley Publishing Company Inc., 1989.

[12] D. C. Walters and G. B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading," *IEEE Trans. on PWRS*, vol. 8, no. 3, pp. 1325–1332, 1993.

[13] P. H. Chen and H. C. Chang, "Large-scale economic dispatch by genetic algorithm," *IEEE Trans. on PWRS*, vol. 10, no. 4, pp. 1919–1926, 1995.

[14] D. Dasgupta and D. R. McGregor, "Thermal unit commitment using genetic algorithms," *IEE Proc.—Gener. Transm. Distrib.*, vol. 141, no. 5, pp. 459–465, 1994.

[15] G. B. Sheble, T. T. Maifeld, K. Brittig, and G. Fahd, "Unit commitment by genetic algorithm with penalty methods and a comparison of Lagrangian search and genetic algorithm-economic dispatch algorithm," *Int. Journal of Electric Power & Energy Systems*, vol. 18, no. 6, pp. 339–346, 1996.

[16] K. Iba, "Reactive power optimization by genetic algorithm," *IEEE Trans. on PWRS*, vol. 9, no. 2, pp. 685–692, 1994.

[17] K. Y. Lee, X. Bai, and Y. M. Park, "Optimization method for reactive power planning using a genetic algorithm," *IEEE Trans. on PWRS*, vol. 10, no. 4, pp. 1843–1850, 1995.

[18] K. Y. Lee and F. F. Yang, "Optimal reactive power planning using evolutionary algorithms: A comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming," *IEEE Trans. on PWRS*, vol. 13, no. 1, pp. 101–108, 1998.

[19] R. Dimeo and K. Y. Lee, "Boiler–Turbine control system design using a genetic algorithm," *IEEE Trans. on Energy Conversion*, vol. 10, no. 4, pp. 752–759, 1995.

[20] Y. Zhao, R. M. Edwards, and K. Y. Lee, "Hybrid feedforward and feedback controller design for nuclear steam generators over wide range operation using genetic algorithm," *IEEE Trans. on Energy Conversion*, vol. 12, no. 1, pp. 100–106, 1997.

Jong-Bae Park received B.S., M.S., and Ph.D. degrees in electrical engineering from Seoul National University in 1987, 1989, and 1998, respectively. For 1989–1998, he worked as a Researcher of Korea Electric Power Corporation, and since 1998 he has been an Assistant Professor of electrical engineering at Anyang University, Korea. His research interests are power system planning, optimization, and economic studies.

Young-Moon Park was born in Masan, Korea on Aug. 20 1933. He received his B.S., M.S. and Ph.D. degrees from Seoul National University in 1956, 1959 and 1971, respectively. For 1959–1998, he was with Seoul National University where he is currently a Professor Emeritus of electrical engineering. From 1998, he has also been an Adjunct Professor of electrical engineering at the Pennsylvania State University, U.S.A. His interests include power systems operation and planning, expert systems and artificial intelligence application to power systems.

Jong-Ryul Won received B.S., M.S., and Ph.D. degrees in electrical engineering from Seoul National University in 1993, 1995, and 1998, respectively. Since 1998, he has been a Researcher of Korea Electric Power Research Institute. His research interests are power system planning, and optimization.

Kwang Y. Lee received B.S. degree in electrical engineering from Seoul National University, Korea, in 1964, M.S. degree in electrical engineering from North Dakota State, Fargo in 1968, and Ph.D. degree in Systems Science from Michigan State, East Lansing in 1991. He has been with Michigan State, Oregon State, Univ. of Houston, and the Pennsylvania State University, where he is a Professor of electrical engineering and Director of Power Systems Control Laboratory. His interests include control and intelligent systems and their applications to power plant and power system control, operation and planning.