

# 유전알고리즘을 이용한 발전계통의 보수계획 수립

論 文

48A-5-9

## Maintenance Scheduling using a Genetic Algorithm with New Crossover Operators

鄭正源\* ·金正益\*\*

(Jung-Won Jung, Jung-Ik Kim)

**Abstract** - The Maintenance scheduling is one of the mid-term scheduling problems of power systems. There have been many methods for this problem, but there is no effective way to treat all the generators simultaneously. In this paper, we apply a genetic algorithm(GA) to the maintenance scheduling problem. We proposed new crossover operators(Box type crossover) to improve searching ability of GA. Satisfactory results are obtained by GA with the proposed crossover operators.

**Key Words** : Generator Maintenance Scheduling, Genetic Algorithm, Box type crossover

### 1. 서 론

전력계통에서 발전기 보수계획은 경제적 운용과 공급 신뢰도 측면에서 매우 중요한 문제이다. 전력 수요의 지속적인 증가 속에서 양질의 전력공급을 위한 신뢰도 확보는 전력계통의 중요한 과제이다. 현재 우리 나라의 전력계통도 전력수요의 증가와 함께 대규모화되었고, 발전계통의 신뢰도를 보충할 수 있는 보수계획수립도 더욱 어려워지고 있다.

이상적인 발전기 보수계획은 발전비용과 발전계통의 신뢰도를 함께 고려할 수 있는 것이다. 보수계획 수립시 발전비용과 신뢰도를 같이 고려하기 위해서는 발전계통의 신뢰도를 비용으로 나타낼 수 있어야 하나 이러한 표현 자체가 어려운 문제이기 때문에 보수계획 문제 수립시에는 발전비용 또는 신뢰도를 단독 목적함수로 두고 최적화 하는 것이 일반적이다.[1] 발전 비용을 목적함수로 사용하면 많은 근사화 과정이 필요하거나 계산량이 많은 시뮬레이션을 필요로 한다. 또한 전원 계획이 적절히 수립되어 있어 적정 규모의 설비를 가지고 있는 경우에는 보수계획의 목적은 전력공급의 신뢰도를 최상으로 유지하는데 있다고 할 수 있다. 따라서, 본 논문에서는 보수 계획의 목적함수로서 신뢰도를 사용했다. 신뢰도 목적함수는 예비력, 예비율 등의 결정적 지수와 공급지장확률(LOLP), 공급지장에너지의 기대치 등의 확률적 지수로 나눌 수 있다. 결정적 지수를 목적함수로서 사용하면 보수계획문제를 단순화시킬 수 있으나, 발전기의 고장정지와 같은 확률

적 변수를 다룰 수 없다. 따라서, 확률적 지수인 공급지장 확률 또는 공급지장 에너지의 기대치를 평활화하거나 최소화하도록 목적함수를 설정하는 것이 일반적이다. 현재의 발전계통에서는 발전비용 보다는 공급 신뢰도의 중요성이 가중되고 있으므로 본 논문에서는 연간 공급지장확률의 최소화를 목적함수로 선택하였다.

보수계획문제를 해결하기 위해 여러 가지 기법들이 적용되었다. 과거에는 정수계획법(integer programming), 동적계획법(dynamic programming), 분지한정법(branch and bound method)등이 고려되었다. 정수계획법으로서는 전력계통에 내재하는 불확실성을 고려하기 힘들고 동적계획법과 분지한정법 등은 기억용량 및 계산시간 제약으로 모든 발전기를 동시에 고려할 수 없다. 최근에는 보수계획문제에 시뮬레이티드 어닐링(simulated annealing)과 유전 알고리즘(genetic algorithm) 등의 적용이 시도되고 있다.[2][3]

보수계획문제는 조합의 최적화 문제이며, 무한대에 가까운 조합의 개수를 갖는다. 본 연구에서는 보수계획문제의 최적화 기법으로서 유전 알고리즘을 적용하였다. 최적화 과정에 자연 유전자와 자연 선택을 모방한 유전 알고리즘은 전역 탐색을 통해 최적해를 구할 수 있는 기법이며, 구속조건을 갖는 최적화 문제나 조합의 최적화 문제를 다룰 수 있는 적절한 기법이다.

유전 알고리즘의 효율은 교배연산자에 의해 많은 영향을 받는다. 본 논문에서는 유전 알고리즘의 효율을 높이기 위해 기존의 교배연산자인 선형 교배연산자(linear crossover)와 컨벡스 교배연산자(convex crossover)의 절충형인 새로운 박스 교배연산자(box crossover) 및 이를 개선한 이동 박스 교배연산자(shifted box crossover), 확장 박스 교배연산자(enlarged box crossover)를 제안하였다.

\* 正 會 員 : 慶星大 電氣電子컴퓨터工學部 副教授 · 工博

\*\* 準 會 員 : 慶星大 電氣工學科 碩士課程

接受日字 : 1999년 2월 11일

最終完了 : 1999년 4월 7일

## 2. 목적 함수의 선정

### 2.1 LOLP의 계산

본 논문에서는 발전계통의 보수계획문제의 목적 함수로서 연간 공급지장확률(LOLP; loss of load probability)를 최소화하도록 설정하였다.

LOLP의 계산을 위해서 다음과 같이 가정하였다.[4]

- 각 발전기는 고장 및 운전의 두가지 상태로 설정하였다.
- 각 발전기의 고장정지는 서로 독립적인 것으로 가정한다.

LOLP의 계산에는 등가부하지속곡선(ELDC; equivalent load duration curve)을 이용하였다.  $i$  발전기까지의 고장정지를 고려한 등가부하지속곡선은  $i-1$  발전기까지의 고장정지 영향을 고려한 등가부하지속곡선에  $i$  번째 발전기의 운전상태를 상승 적분하면 구해진다.

$$F_i(x) = p_i \cdot F_{i-1}(x) + q_i \cdot F_{i-1}(x - C_i) \quad (1)$$

여기서,  $p_i$  :  $i$  발전기의 가동율( $1 - q_i$ )

$q_i$  :  $i$  발전기의 고장정지율

$C_i$  :  $i$  발전기의 발전용량

$F_i$  :  $i$  발전기까지의 고장정지를 고려한 등가부하지속곡선

모든 발전기의 고장정지를 고려한 등가부하지속곡선의 시설 용량점에 대응하는 값은 등가부하가 시설용량을 초과할 확률, LOLP를 나타낸다.

$$LOLP = F_n \left( \sum_i C_i \right) \quad (2)$$

여기서,  $n$  = 발전기의 수

따라서, 발전기의 보수정지를 고려한  $j$ 주의 LOLP는 다음으로 표시된다.

$$LOLP^j = F^j \left( \sum_{i \in I(j)} C_i \right) \quad (3)$$

여기서,  $I(j)$ 는  $j$  주의 보수정지중이지 않은 발전기의 집합,  $LOLP^j$ 는  $j$ 주의 LOLP이고,  $F^j$ 는  $j$ 주의 보수정지중인 발전기를 제외한 모든 발전기의 고장정지를 고려한 등가부하지속곡선이다. 따라서, 목적함수인 연간 LOLP는 주별 LOLP를 합하여 구할 수 있다.

$$\text{연간 LOLP} = \sum_{j=1}^{52} LOLP^j \quad (4)$$

### 2.1 기술자 제약조건(crew constraint)을 고려한 목적함수

주별 보수 물량은 발전계통을 보수·정비를 하는 인력에 의해 한정된다. 따라서, 보수계획 수립에 있어서 기술자 제약조건을 고려하였다. 이 제약조건은 LOLP와 함께 벌점(penalty)

항으로서 목적함수에 포함시켜서 고려하였다. 이것을 다음과 같이 표현할 수 있다.

$$\text{objective} = \sum_{j=1}^{52} (LOLP^j + (M_j - M_j) \cdot f(x)) \quad (5)$$

여기서,  $M_j$  :  $j$ 주의 보수 정지 중인 발전기 수

$M_j$  : 최대 보수 정지할 수 있는 발전기 수

$f(x)$  : 세대수에 따른 벌점의 정도를 조절하는 함수

## 3. 보수계획에 대한 유전 알고리즘의 적용

본 논문에서 제시된 유전 알고리즘을 사용한 보수계획의 순서도를 그림 1에 나타내었다. 그림 1을 살펴보면, 처음 발전계통의 자료를 입력받는다. 이 자료에는 각 발전기의 발전 용량, 고장 정지율, 보수정지기간, 보수가능기간이 포함되어 있다. 이 자료로부터 보수가 고정된 발전기, 보수를 하지 않는 발전기, 보수기간을 결정해야 하는 발전기를 분류한다. 분류된 발전기 중 보수정지기간을 결정해야 하는 발전기의 보수기간 시작점을 유전 알고리즘의 개체로써 표시한다. 각 개체들은 연간 LOLP 계산을 통해 적합도 계산을 하고, 유전 알고리즘의 연산을 통해 새로운 개체들을 생성한다. 최종 개체 중 최고의 적합도를 가진 개체를 출력한다.

### 3.1 개체

본 논문에서 사용된 유전알고리즘의 개체는 실수로 발생시켰다. 각각의 개체들은 각 발전기의 보수가능기간 안에서 발생시켰고, 발전기 보수기간의 시작점으로 나타내었다. 즉, 아래와 같이 발생시켰다.

$$u_i^j = \lambda_i^j (M_{end}^i - M_{start}^i - M_{duration}^i + 1.999) \quad (6)$$

여기서,  $u_i^j$  ; 개체  $j$ 의 발전기  $i$ 의 보수기간의 시작점

$M_{start}^i$  ; 발전기  $i$ 의 보수가능기간의 시작점

$M_{end}^i$  ; 발전기  $i$ 의 보수가능기간의 끝점

$M_{duration}^i$  ; 발전기  $i$ 의 보수기간

$\lambda_i^j$  ; 0과 1사이의 임의의 수

$u_i^j$ 는  $j$ 개체(individual)의  $i$  번째 발전기의 보수시작시점을 표시하기 위한 변수이다. 만일  $i$  발전기의 보수가능시기가 10주에서 30주 사이이고 이 발전기의 보수기간이 5주이라면 이 발전기의 보수시작 점은 10주에서 26주까지 17개의 가능한 시작점을 갖는다.(30-10-5+2). 이 때  $u_i^j$ 는 0~17사이의 값을 갖는 실수이고 연간 LOLP 계산시에는 실수  $u_i^j$ 를 정수로 변환시켜서 보수기간을 산출하여 LOLP 계산에 적용하였다.

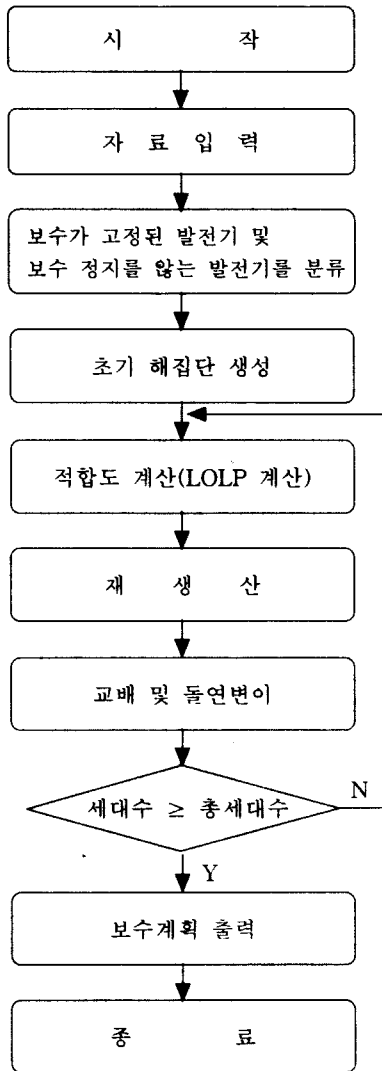


그림 1. 유전알고리즘을 이용한 보수계획 순서도  
Fig 1. Flowchart of a maintenance scheduling using GA

3.2 적합도 함수와 재생산

적합도 함수는 목적함수를 최소화하면서 최대화되도록 설정하였으며, 식 (7)으로 표현할 수 있다.

$$Fitness = \left[ \frac{\beta}{a + objective} \right] \quad (7)$$

여기서,  $\alpha$ 와  $\beta$ 는 적합도 함수를 조절하기 위한 매개변수이며, 목적함수는 LOLP만을 고려한 것과 LOLP와 기술자 제약조건을 같이 고려한 경우 등 두 가지로 나눌 수 있다.

재생산은 룰렛 휠(roulette wheel)을 사용하였다.

3.3 교배

본 논문에서는 유전알고리즘의 기능을 향상시키기 위해 새로운 박스형 교배연산자를 제시하였다. 기존의 교배연산자인 산술연산자(컨벡스 교배연산자, 선형 교배연산자)와 비교하였

다.[5][6]

3.3.1 산술 교배연산자[5][6]

산술 교배연산자는 두 벡터의 일차결합으로 정의된다. 먼저 컨벡스 교배연산자를 살펴보면, 두 개체  $S'_u$ 와  $S'_v$ 를 교배시키면 자손  $S'^{+1}_u$  및  $S'^{+1}_v$ 은

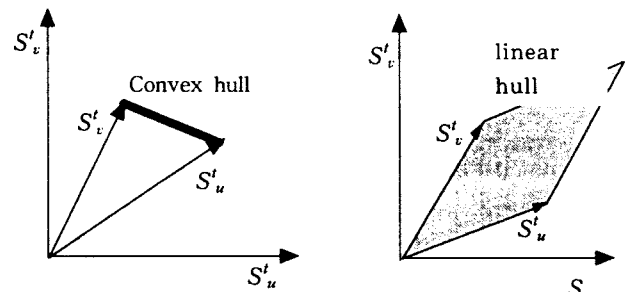
$$\begin{aligned} S'^{+1}_u &= a \cdot S'_v + (1-a) \cdot S'_u \\ S'^{+1}_v &= a \cdot S'_u + (1-a) \cdot S'_v \end{aligned} \quad (8)$$

로 나타난다. 난수  $a$ 는  $a \in [0, 1]$ 이다. 이를 그림으로 나타내면 그림 2.(a)와 같이 새로운 개체  $S'^{+1}_u$ 와  $S'^{+1}_v$ 는 부모인  $S'_u$ 와  $S'_v$ 사이의 직선 영역에서 발생된다.

선형 교배연산자[6]는 식 (9)로 표현된다.

$$\begin{aligned} S'^{+1}_u &= a_1 \cdot S'_v + a_2 \cdot S'_u \\ S'^{+1}_v &= a_1 \cdot S'_u + a_2 \cdot S'_v \end{aligned} \quad (9)$$

여기서, 난수  $a_1, a_2$ 는  $0 \leq a_1 + a_2 \leq 2$ 이고  $a_1, a_2 \in [0, 1]$ 이다. 선형교배연산자의 해 탐색영역을 그림 2. (b)에 나타내었다.



(a) 컨벡스 교배연산자 (a)Convex crossover  
(b) 선형 교배연산자 (b) Linear crossover

그림 2. 산술 교배연산자의 탐색 영역

Fig 2. Illustration showing convex hull and linear hull

3.3.2 박스형 교배연산자

본 논문에서 제시된 교배연산자는 박스 교배연산자, 이동 박스 교배연산자, 확장 박스 교배연산자이다. 먼저 박스 교배연산자는 식 (10)으로 정의할 수 있다. 개체  $S'_u = \langle u'_1, \dots, u'_m \rangle$ 와  $S'_v = \langle v'_1, \dots, v'_m \rangle$ 을 교배시키면 아래와 같이 자손  $S'^{+1}_u = \langle u'^{+1}_1, \dots, u'^{+1}_m \rangle$  및  $S'^{+1}_v = \langle v'^{+1}_1, \dots, v'^{+1}_m \rangle$ 이 구해진다.

$$\begin{aligned}
 u_i^{t+1} &= (v_i^t - u_i^t) \cdot \lambda_i + u_i^t \\
 v_i^{t+1} &= (u_i^t - v_i^t) \cdot \gamma_i + v_i^t
 \end{aligned}
 \tag{10}$$

여기서,  $\lambda_i, \gamma_i \in [0, 1]$ 이다.

박스 교배연산은 컨벡스 교배연산과는 달리 부모의 개체내의 각 변수 각각에 대하여 난수를 발생시켜 자손을 생성한다. 즉, 자손  $S_u^{t+1}$  및  $S_v^{t+1}$ 의  $i$  번째 발전기의 보수정지에 대한 변수  $u_i^{t+1}$  및  $v_i^{t+1}$ 는 부모의 대응 변수  $u_i^t$  및  $v_i^t$ 사이의 값으로 생성된다. 따라서, 박스 교배연산자에 의해 생성된 자손의 범위는 그림3의 (a)와 같이 부모인  $S_u^t$  와  $S_v^t$ 의 끝점을 박스의 대각 모서리로 두는 영역이 된다. 기존의 산술 교배연산자와 비교하면 컨벡스 교배연산자보다는 자손 생성 범위가 넓고, 선형 교배연산자보다는 좁다. 컨벡스 교배연산자에 의한 자손 생성 범위는 그림2의 (a)와 같이 직선 상에 있어 최적해 근방에서는 유리할 수 있으나 국부 수렴의 가능성이 높고, 선형 교배연산자는 탐색영역을 확대하는 의미는 있으나 수렴성이 떨어질 가능성이 높다. 박스 교배연산자는 두 가지를 절충한 형태의 연산자라고 할 수 있다.

또한 본 논문에서는 박스 교배연산자의 탐색 영역을 이동하거나 확장한 이동 박스 교배연산자와 확장 박스 교배연산자도 제안한다. 이동 박스 교배연산자는 룰렛 휠을 통해 선택된 두 개체의 적합도 함수를 비교해서 적합도가 큰 쪽으로 탐색 영역을 이동시킨 것이다. 이는 선택된 두 개체 중 적합도가 낮은 개체 보다 적합도가 높은 개체 근처에서 우수한 자손이 나올 수 있는 개연성이 높다는 것을 반영한다.

이동 박스 교배는 다음과 같이 정의할 수 있다.  $Fitness(S_u^t) < Fitness(S_v^t)$ 라고 하면  $S_v^t$  방향으로의 이동벡터를 다음으로 둔다.

$$d = \tau \cdot (S_v^t - S_u^t) \tag{11}$$

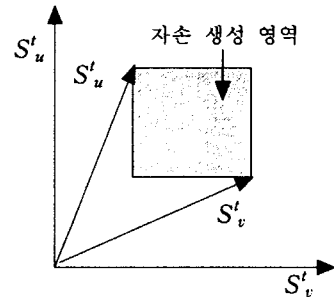
여기서,  $\tau$ 는 탐색영역의 이동 정도를 나타내는 매개변수이다.

그리고, 부모인  $S_u^t$  와  $S_v^t$ 를  $d$  만큼 이동시킨  $S_u^{t'}$  와  $S_v^{t'}$ 를 다음으로 구하고,

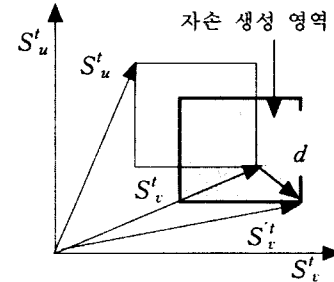
$$\begin{aligned}
 S_u^{t'} &= S_u^t + d \\
 S_v^{t'} &= S_v^t + d
 \end{aligned}
 \tag{12}$$

$S_u^t$  와  $S_v^t$  를  $d$  만큼 이동시킨  $S_u^{t'}$  와  $S_v^{t'}$  에서 박스교배연산에 의해 자손  $S_u^{t+1}$  및  $S_v^{t+1}$  를 구한다.

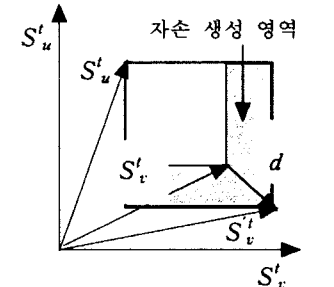
확장 박스 교배연산자는 선택된 두 개체의 적합도 함수를 비교해서 적합도 함수가 큰 쪽으로 탐색영역을 확장한 것이다.  $Fitness(S_u^t) < Fitness(S_v^t)$  라고 하면  $S_v^t$  방향으로의 이동벡터  $d$  를 이동박스 교배에서와 같이 구한다. 적합도가 큰  $S_v^t$  만큼  $d$  만큼 이동시킨  $S_v^{t'}$  를 구하고  $S_u^t$  와  $S_v^{t'}$  를 대상으로 박스교배연산에 의해 자손  $S_u^{t+1}$  및  $S_v^{t+1}$  를 구한다.



(a) 박스 교배연산자  
(a) Box crossover



(b) 이동 박스 교배연산자  
(b) Shifted box crossover



(c) 확장 박스 교배연산자  
(c) Enlarged box crossover

그림 3. 박스 교배연산자, 이동 박스 교배연산자, 확장 박스 교배연산자의 자손 생성 영역

Fig 3. Illustration showing offspring areas by box crossover, shifted box crossover, and enlarged box crossover

### 3.4 돌연변이

돌연변이는 불균등 돌연변이(non-uniform mutation)를 사용하였다.[5][6] 주어진 부모  $S$  에 대하여,  $u_k$  가 선택되어지면 자손은  $S = [u_1, \dots, u_k', \dots, u_n]$ 으로 주어진다.  $u_k'$ 는 식 (13)에 의해서 결정된다.

$$u_k' = \begin{cases} u_k + \Delta(t, u_k^U - u_k) \\ u_k - \Delta(t, u_k - u_k^L) \end{cases}
 \tag{13}$$

여기서,  $u_k^U$ 와  $u_k^L$ 는  $x_k$  에 대한 상한치와 하한치이다. 어떤 개체에 대한 돌연변이 적용여부 및  $u_k'$ 의 선택은 난수에 의한다. 함수  $\Delta(t, y)$ 는 식 (14)과 같이  $[0, y]$ 의 값을 가지고,  $t$ (세대수)가 증가

할수록 0에 접근한다. 이러한 성질은 연산자 탐색 작업이 초기에는 비교적 넓은 범위를 균등하게 행해지다가, 점차 지역적이 되도록 해준다.

$$\Delta(t, y) = y \cdot r \cdot \left(1 - \frac{t}{T}\right)^b \quad (14)$$

여기서,  $r \in [0, 1]$ 이고,  $t$ 는 세대수,  $T$ 는 총 세대수,  $b$ 는 불균등 정도를 결정하는 매개변수이다.

#### 4. 사례연구

##### 4.1 박스형 교배연산자와 산술 교배연산자의 비교

본 논문에서 제시된 박스형 교배연산자의 효율을 알아보기 위해서 알려진 함수에 대해 산술 교배연산자와 비교하였다. 비교한 함수는 Colville 함수, GoldsteinP 함수 및 Floudas 함수이다.[5]

· Colville 함수

$$f(x) = 100(x_1^2 - x_2)^2 + 90(x_4 - x_3)^2 + (1 - x_3)^2 + 0.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1) \quad (15)$$

· GoldsteinP 함수

$$f(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \cdot [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)] \quad (16)$$

· Floudas 함수

$$f(x) = -5 \sin(x_1) \sin(x_2) \sin(x_3) \sin(x_4) \sin(x_5) - \sin(5x_1) \sin(5x_2) \sin(5x_3) \sin(5x_4) \sin(5x_5) \quad (17)$$

유전 알고리즘에 사용된 매개 변수를 다음과 같이 설정하였다.

- 개체수=50, 세대수=500
- 교배확률=0.8, 돌연변이확률=0.02
- $\tau=0.5, b=5$

여기서,  $\tau$ 는 박스형 교배연산자에서 탐색영역의 이동 또는 확장을 조절하기 위한 매개 변수이고,  $b$ 는 불균등 돌연변이에서 불균등 정도를 결정하는 매개변수이다.

그림 4, 5, 6에 각각의 함수에 대한 최대 적합도 추이를 나타내었다.

이상의 사례연구에서는 제시된 교배연산자와 기존의 교배연산자의 탐색 성능을 비교하였다. 각각의 결과들은 이동 박스 교배연산자와 확장 박스 교배연산자를 적용했을 때 빠른 수렴성과 좋은 적합도를 보여주고 있다.

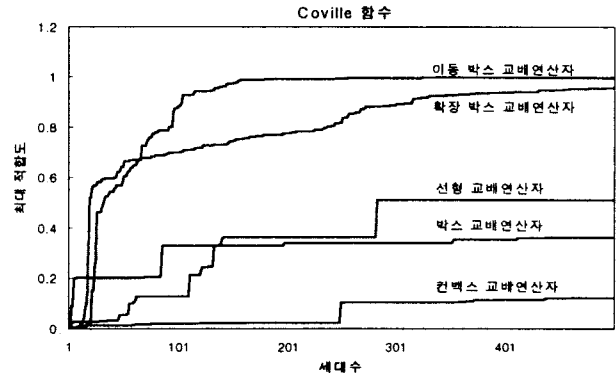


그림 4. Coville 함수의 적합도

Fig 4. Trends of best fitness value of the Coville function

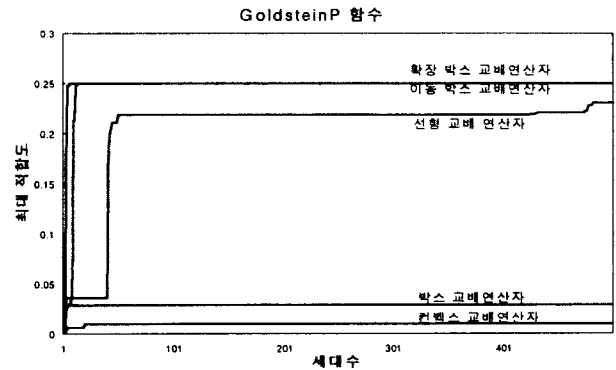


그림 5. GoldsteinP의 최대 적합도

Fig 5. Trends of best fitness value of the GoldsteinP function

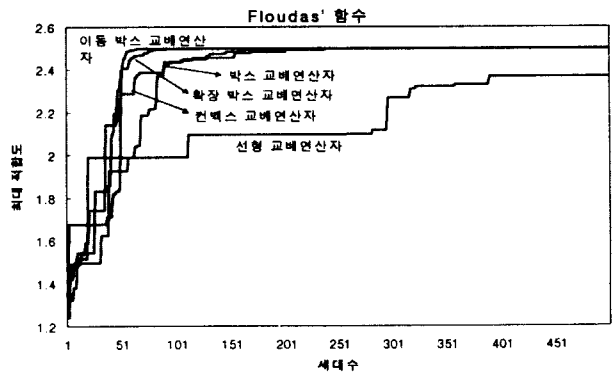


그림 6. Floudas' 함수의 최고 적합도

Fig 6. Trends of best fitness value of the Floudas' function

4.2 발전계통의 보수계획

본 논문에서의 보수계획 수립에 사용된 자료는 참고문헌 [4]에 사용된 발전기 자료를 사용하였다. 참고문헌의 자료는 발전기가 59대, 최대부하가 20,438 MW, 시설용량이 23,417 MW의 계통이다. 표 3에 발전기 자료를 나타내었다.

표 3. 발전기 자료[4]  
Table 3. Data of generators

발전기명	용량 (MW)	F.O.R	보수일 수(주)	발전기명	용량 (MW)	F.O.R	보수일 수(주)
HYD	1445	0.0	0	ther#20	200	0.030	7
PMP	1000	0.0	0	ther#21	280	0.073	7
nuc #1	587	0.082	10	ther#22	280	0.073	7
unc #2	650	0.082	10	ther#23	560	0.041	5
nuc #3	950	0.082	10	ther#24	560	0.041	5
nuc #4	950	0.082	10	ther#25	500	0.030	7
nuc #5	678	0.062	10	ther#26	500	0.030	7
nuc #6	950	0.082	10	ther#27	200	0.049	5
nuc #7	950	0.082	10	ther#28	200	0.049	5
nuc #8	950	0.082	10	ther#29	200	0.069	5
nuc #9	950	0.082	10	ther#30	300	0.086	5
ther#1	138	0.045	5	ther#31	50	0.066	5
ther#2	250	0.045	5	ther#32	50	0.066	5
ther#3	200	0.072	5	ther#33	75	0.040	5
trer#4	200	0.072	5	ther#34	60	0.068	5
ther#5	200	0.072	5	ther#35	60	0.068	5
ther#6	400	0.069	5	ther#36	105	0.062	6
ther#7	400	0.069	5	ther#37	105	0.062	6
ther#8	400	0.069	5	ther#38	162	0.004	5
ther#9	250	0.029	8	ther#39	162	0.004	5
ther#10	250	0.029	8	ther#40	300	0.004	6
ther#11	325	0.086	8	ther#41	300	0.040	6
ther#12	325	0.086	8	ther#42	270	0.040	6
ther#13	350	0.039	5	ther#43	55	0.060	5
ther#14	350	0.039	5	ther#44	940	0.050	0
ther#15	350	0.039	5	ther#45	620	0.050	0
ther#16	350	0.039	5	ther#46	400	0.050	0
ther#17	125	0.054	5	ther#47	300	0.050	0
ther#18	200	0.041	5	ther#48	300	0.050	0
ther#19	200	0.030	7				

본 연구에서는 두 가지 경우에 대해서 보수계획을 수립하였다. 첫 번째는 기존의 보수계획의 결과[4]와 비교하기 위해 기술자 제한조건을 고려하지 않은 경우와 두 번째는 기술자 제한조건을 고려하는 경우로 나누어서 보수계획을 수립하였다. 기존의 방법은 최적화 기법으로서는 분지한정법을 사용하였으며, 발전기 전부를 변수로 처리하지 못하기 때문에 완화법과 같이 발전기를 그룹으로 나누어 순차적으로 최적화하는 방법이다.

기술자 제약 조건을 적용하지 않은 보수계획문제의 목적함수는 다음과 같이 설정하였다.

$$Fitness = \left[ \frac{\beta}{\alpha + objective} \right] \tag{18}$$

$$objective = \sum_{j=1}^{59} LOLP^j \tag{19}$$

보수계획에 사용된 유전알고리즘의 매개변수는 다음과 같이 설정하였다.

- 개체수=100, 세대수=300
- $\alpha=1, \beta=100$
- 교배확률=0.8, 돌연변이확률=0.01
- $\tau=0.5, b=5$

여기서,  $\alpha, \beta$ 는 적합도 함수를 조절하기 위한 매개 변수이고,  $\tau$ 는 박스 교배연산자에서 탐색영역의 이동 또는 확장을 조절하기 위한 매개 변수이고,  $b$ 는 불균등 돌연변이에서 불균등 정도를 결정하는 매개변수이다.

표 4에 유전 알고리즘에서 구한 연간 LOLP와 기존 방법 [4]의 연간 LOLP를 비교하였다. 유전 알고리즘의 연간 LOLP는 각 초기개체에서 최종 연간 LOLP의 평균값이다. 보수계획문제에 유전알고리즘을 적용했을 때의 결과가 분지한정법에서 얻은 결과보다 우수하다는 것을 알 수 있다. 분지한정법 또는 동적계획법 등은 보수계획에 적용되는 실용적인 방법이나 대상 발전기 수가 커질수록 지수 함수적으로 계산 시간이 늘어나기 때문에 실제 규모의 보수계획 수립에서는 어려움이 있고, 전체발전기를 그룹으로 나누어 고려하는 한 국지적 최소점에 빠질 가능성이 크다. 반면 유전알고리즘은 비록 모든 가능해를 탐색하지는 않으나 전역적 탐색이 가능한 알고리즘이며 한 개체가 전체 발전기의 보수정지 정보를 갖고 있으므로, 기존의 방법보다는 전역 최적해에 가까운 해를 찾을 수 있기 때문이다.

표 4. 연간 LOLP 비교  
Table 4. Comparison of LOLP

	유전 알고리즘				분지한정법 [4]
	이동 박스 교배연산자	확장 박스 교배연산자	컨베ックス 교배연산자	선형 교배연산자	
연간 LOLP	45.24	45.15	47.68	48.73	77.20

그림 7에 교배 연산자에 대해 연간 LOLP를 비교하였다. 그림 7에서 나타난 이동 박스 교배연산자와 확장 박스 교배연산자를 적용한 경우에 가장 우수한 연간 LOLP를 보여주고 있으며, 다른 교배연산자를 적용한 경우에 비해 보다 빠른 수렴성을 보여주고 있다. 표 5에 확장 박스 교배연산자를 적용한 보수계획의 결과를 나타내었다.

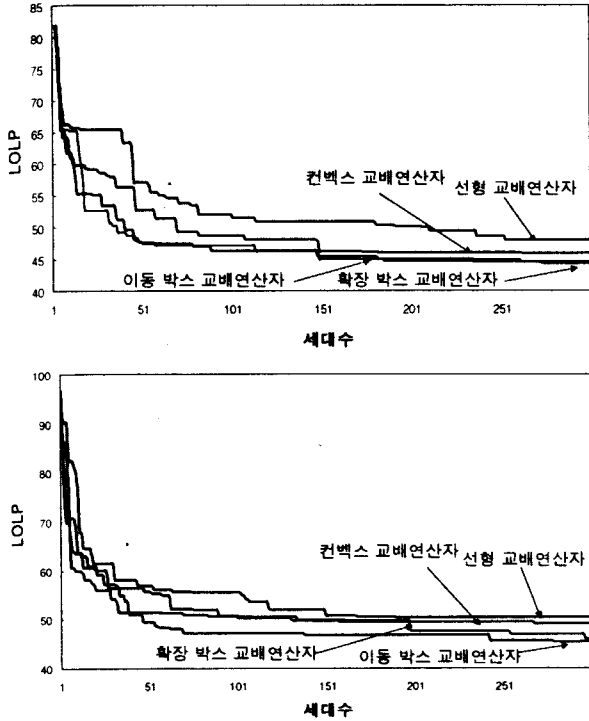


그림 7. 연간 LOLP 비교  
Fig. 7. Comparison of LOLP

두 번째 사례로서 보수계획문제에 기술자 제약조건을 고려하였다. 이때 적합도 함수는 다음과 같이 설정하였다.

$$Fitness = \left[ \frac{\beta}{\alpha + objective} \right] \quad (20)$$

$$objective = \sum_{j=1}^{50} (LOLP_j + (M_j - M_j) \cdot f(x)) \quad (21)$$

여기서,  $M_j$  : j주의 보수정지 중인 발전기의 수  
 $M_j$  : 보수정지 할 수 있는 발전기의 최대수  
 $f(x)$  : 세대수에 따른 벌점의 정도를 조절하는 함수

$M_j$ 는 7,  $f(x)$ 는  $200 \cdot \text{Tanh}(x)$ 로 설정하였다.  $200 \cdot \text{Tanh}(x)$ 는 세대수가 증가함에 따라 0에서 200까지 변화한다. 이는 제약 조건이 세대가 증가 할수록 만족하도록 한 것이다. 유전알고리즘의 매개 변수의 설정치 중 개체수를 200으로 설정하고 나머지 매개 변수들은 첫 번째 사례와 동일하게 설정하였다.

표 6.에 나타난 바와 같이 이동 박스 교배연산자에서 가장 우수한 연간 LOLP값을 보여주고 있다. 선형 교배연산자를 적용한 경우에는 기술자 제약조건을 만족하는 연간 LOLP를 구할 수 없었다. 표 7.에 이동 박스 교배연산자를 적용한 경우의 보수계획의 결과를 나타내었다.

표 5. 확장 박스 교배연산자를 이용한 보수계획의 결과  
Table 5. Maintenance scheduling with enlarged box

crossover

	10	20	30	40	50
HYD	-----	-----	-----	-----	-----
PMP	-----	-----	-----	-----	-----
nuc #1	XXXXXXXXXX	-----	-----	-----	-----
unc #2	XXXXXXXXXX	-----	-----	-----	-----
nuc #3	-----	-XXXXXXXXX	X-----	-----	-----
nuc #4	-----	XXXXX	XXXXX	-----	-----
nuc #5	-----	-----	-----	-----	-----
nuc #6	-----	XXXXXXXXXX	-----	-----	-----
nuc #7	XXXXXXXXXX	-----	-----	-----	-----
nuc #8	XXXXXXXXXX	-----	-----	-----	-----
nuc #9	-----	XXXX	XXXXXX	-----	-----
ther#1	-----	XXXXX	-----	-----	-----
ther#2	-----	-----	-----	-----	-----
ther#3	-----	-----	-----	-----	-----
trr#4	-----	-----	-----	-----	-----
ther#5	-----	-----	-----	-----	-----
ther#6	-----	-----	-----	-----	-----
ther#7	-----	-----	-----	-----	-----
ther#8	-----	-----	-----	-----	-----
ther#9	-----	-----	-----	-----	-----
ther#10	-----	-----	-----	-----	-----
ther#11	-----	-----	-----	-----	-----
ther#12	-----	-----	-----	-----	-----
ther#13	-----	-----	-----	-----	-----
ther#14	-----	-----	-----	-----	-----
ther#15	-----	-----	-----	-----	-----
ther#16	-----	-----	-----	-----	-----
ther#17	-----	-----	-----	-----	-----
ther#18	-----	-----	-----	-----	-----
ther#19	-----	-----	-----	-----	-----
ther#20	-----	-----	-----	-----	-----
ther#21	-----	-----	-----	-----	-----
ther#22	-----	-----	-----	-----	-----
ther#23	-----	-----	-----	-----	-----
ther#24	-----	-----	-----	-----	-----
ther#25	-----	-----	-----	-----	-----
ther#26	-----	-----	-----	-----	-----
ther#27	-----	-----	-----	-----	-----
ther#28	-----	-----	-----	-----	-----
ther#29	-----	-----	-----	-----	-----
ther#30	-----	-----	-----	-----	-----
ther#31	-----	-----	-----	-----	-----
ther#32	-----	-----	-----	-----	-----
ther#33	-----	-----	-----	-----	-----
ther#34	-----	-----	-----	-----	-----
ther#35	-----	-----	-----	-----	-----
ther#36	-----	-----	-----	-----	-----
ther#37	-----	-----	-----	-----	-----
ther#38	-----	-----	-----	-----	-----
ther#39	-----	-----	-----	-----	-----
ther#40	-----	-----	-----	-----	-----
ther#41	-----	-----	-----	-----	-----
ther#42	-----	-----	-----	-----	-----
ther#43	-----	-----	-----	-----	-----
ther#44	-----	-----	-----	-----	-----
ther#45	-----	-----	-----	-----	-----
ther#46	-----	-----	-----	-----	-----
ther#47	-----	-----	-----	-----	-----
ther#48	-----	-----	-----	-----	-----

표 6. 기술자 제한조건을 고려한 연간 LOLP 비교  
Table 6. Comparison of LOLP with crew constraint

	유전 알고리즘			
	이동 박스 교배연산자	확장 박스 교배연산자	컨백스 교배연산자	선형 교배연산자
연간 LOLP	53.873	63.370	90.219	95.896

표 7. 기술자 제약조건을 고려한 보수계획 결과  
Table 7. Maintenance scheduling restricted to crew constraint

	10	20	30	40	50
HYD	-----	-----	-----	-----	-----
PMP	-----	-----	-----	-----	-----
nuc #1	-XXXXXXXX	X-----	-----	-----	-----
unc #2	XXXXXXXXX	X-----	-----	-----	-----
nuc #3	-----	-----	-----	-----	-----
nuc #4	-----	-----XXXX	XXXXXX	-----	-----
nuc #5	-----	-----	-----	-----	-----
nuc #6	-----	XXXXXXXXXX	-----	-----	-----
nuc #7	XXXXXXXXXX	-----	-----	-----	-----
nuc #8	-----	-----	-----	-----	-----
nuc #9	-----	-----	-----	-----	-----
ther#1	-----	-----	-----	-----	-----
ther#2	-----	-----	-----	-----	-----
ther#3	-----	-----	-----	-----	-----
ther#4	-----	-----	-----	-----	-----
ther#5	-----	-----	-----	-----	-----
ther#6	-----	-----	-----	-----	-----
ther#7	-----	-----	-----	-----	-----
ther#8	-----	-----	-----	-----	-----
ther#9	-----	-----	-----	-----	-----
ther#10	-----	-----	-----	-----	-----
ther#11	-----	-----	-----	-----	-----
ther#12	-----	-----	-----	-----	-----
ther#13	-----	-----	-----	-----	-----
ther#14	-----	-----	-----	-----	-----
ther#15	-----	-----	-----	-----	-----
ther#16	-----	-----	-----	-----	-----
ther#17	-----	-----	-----	-----	-----
ther#18	-----	-----	-----	-----	-----
ther#19	-----	-----	-----	-----	-----
ther#20	-----	-----	-----	-----	-----
ther#21	-----	-----	-----	-----	-----
ther#22	-----	-----	-----	-----	-----
ther#23	-----	-----	-----	-----	-----
ther#24	-----	-----	-----	-----	-----
ther#25	-----	-----	-----	-----	-----
ther#26	-----	-----	-----	-----	-----
ther#27	-----	-----	-----	-----	-----
ther#28	-----	-----	-----	-----	-----
ther#29	-----	-----	-----	-----	-----
ther#30	-----	-----	-----	-----	-----
ther#31	-----	-----	-----	-----	-----
ther#32	-----	-----	-----	-----	-----
ther#33	-----	-----	-----	-----	-----
ther#34	-----	-----	-----	-----	-----
ther#35	-----	-----	-----	-----	-----
ther#36	-----	-----	-----	-----	-----
ther#37	-----	-----	-----	-----	-----
ther#38	-----	-----	-----	-----	-----
ther#39	-----	-----	-----	-----	-----
ther#40	-----	-----	-----	-----	-----
ther#41	-----	-----	-----	-----	-----
ther#42	-----	-----	-----	-----	-----
ther#43	-----	-----	-----	-----	-----
ther#44	-----	-----	-----	-----	-----
ther#45	-----	-----	-----	-----	-----
ther#46	-----	-----	-----	-----	-----
ther#47	-----	-----	-----	-----	-----
ther#48	-----	-----	-----	-----	-----

5. 결 론

본 논문에서는 연간 LOLP를 최소화하는 보수계획수립문제의 최적해를 도출함에 있어서 대규모 최적화 기법으로서 유전알고리즘을 도입하였으며, 유전알고리즘의 탐색 성능을 향상시키기 위해서 박스 교배연산자, 이동 박스 교배연산자, 확장 박스 교배연산자 등의 새로운 교배연산자를 제시하였다.

제시된 박스형 교배연산자들은 유전 알고리즘의 성능 비교를 위해 몇 가지 함수에 대한 적용을 통해서 기존의 교배연산자를 사용한 유전 알고리즘보다 탐색 능력이 향상되었음을 확인할 수 있었다. 그리고 향상된 탐색 성능을 가진 유전 알고리즘을 통해서 보수계획문제의 최적해를 구할 수 있었다. 유전 알고리즘을 통해 얻은 수치적 결과는 기존의 결과 [4]와의 비교를 통해 보다 우수하다는 것을 확인할 수 있었다.

이 논문은 1997년 한국학술진흥재단의 공모과제 연구비에 의하여 연구되었습니다.

참 고 문 헌

- [1] Z. A. Yamayee, "Maintenance Scheduling : Description, Literature Survey, and Interface with Overall Operations Scheduling", IEEE Transaction on Power Apparatus and System, Vol. PAS-101, No.8, August 1982.
- [2] H. H. Zurn V. H. Quintana, "Generator Maintenance Scheduling via Successive Approximation Dynamic Programming", IEEE Transaction on Power Apparatus and System, Vol. PAS-94, pp.665~671, March/April 1975.
- [3] Aijaz Ahmad & D.P. Kothari, "A review of advances in generator maintenance scheduling", Electric Machine & Power System, Vol 6. pp.373~387, 1998.
- [4] 정 정 원, "분지한정법을 이용한 발전계통의 보수계획수립에 관한 연구 : Trans. KIEE, Vol 44, pp.829~835, 1995
- [5] Zbigniew Michalewicz, "Genetic Algorithms + Data Structures = Evolution Program", Springer-Verlag, 1992
- [6] Mitsuo Gen & Runwei Cheng, "Genetic Algorithms & Engineering Design", WILEY - INTERSCIENCE

저 자 소 개

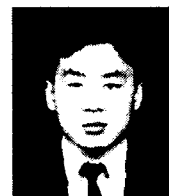


정 정 원(鄭正源)

1960년 11월 8일생. 1983년 서울대학교 전기공학과 졸업. 1985년 동 대학원 전기공학과 졸업(석사). 1991년 동 대학원 전기공학과 졸업(박사). 1985~1991년 한국전력공사 근무. 현재 경성대학교 전기전자컴퓨터공학부 부교수

Tel : (051) 620-0794

E-mail : jjwon@star.kyungsung.ac.kr



김 정 익(金正益)

1971년 6월 20일생. 1997년 경성대학교 물리학과 졸업. 1999년 동 대학원 전기공학과 석사과정.