

☒ 응용논문

## 유전자 알고리즘을 이용한 비선형 모형의 D-최적 실험계획법에 관한 연구

염준근 · 남기성

동국대학교 이과대학 통계학과

### A Study of D-Optimal Design in Nonlinear Model Using the Genetic Algorithm

Joon Keun Yum · Ki Seong Nam

Dept. of Statistics, Dongguk University

#### Abstract

This study has adapted a genetic algorithm for an optimal design for the first time. The models using a simulation are the nonlinear models. Using an genetic algorithm in D-optimal, it is more efficient than previous algorithms to get an object function. Not like other algorithms, without any troublesome restrictions about the initial solution, not falling into a local optimal solution, it's the most suitable algorithm. Also if we use it without any adding experiments, we can use it to find optimal design of experimental condition efficiently.

#### 1. 서론

최적실험계획법(optimal design of experiments) 또는 최적계획법(optimal design)이라 불리는 최적화 문제는 가장 적합한 결과를 얻기 위한 실험의 계획 방법을 의미하는 것으로 해결하고자 하는 실험에서 특정한 요인에 대해 어떤 수준의 관측값을 취할 것인가를 결정하는 것이다. 예를 들어 어떤 도시에서 두 지점간의 최단 거리를 찾거나, 주어진 공정에서 혼합되는 재료의 가장 적합한 비율을 찾거나, 주어진 공간에서의 최적의 해를 찾는 것을 목적으로 할 수 있다. 판정기준(criterion)에 의해 이와 같은

목적에 맞는 목적함수(objective function)를 결정하고, 이를 최적화시키는 독립변수들의 값을 알아내는 것이 최적실험계획법이다.

1970년대 이전까지는 최적실험계획법에 대한 이론들이 많이 연구되었다. 최적실험계획법에 대한 알고리즘은 Fedorov(1972), Mitchell(1974)등에 의해서 많은 발전을 이루어왔고, 1980년대에는 컴퓨터를 이용한 실용적인 면의 시도가 Lundy(1985), Lundy와 Mees(1986), Bohachevsky 등(1986), Haines(1987)에 의해서 이루어졌다.

D-최적실험계획법에서의 접근한 많은 방법들은 수학적인 프로그램 기술에 의해  $|X'X|$ 를 직접적으로 최대화하였다. D-최적계획법에서 이러한 접근들의 고유의 문제(차원 등) 때문에 디자인 문제의 특별한 특징 몇 가지를 고려한 방법의 숫자에 더욱 더 관심을 가졌다.

알고리즘은 세 가지로 요약할 수 있다. 첫 번째는 Fedorov(1969, 1972)에 의해 개발되어졌다. 두 번째 것은 Mitchell과 Miller(1970), 그리고 Wynn(1972)가 각각 별개로 주도했다.(앞으로 Wynn-Mitchell 알고리즘으로 인용함). Fedorov와 Wynn-Mitchell 알고리즘 두 개는 단일 점 변환(single point exchange)알고리즘이다. 반대로 Mitchell(1974) 알고리즘은 여러 개의 점을 반복적으로 더하고 빼는 방법이다. 기타 다른 알고리즘은 기본적인 Fedorov알고리즘에서 변화를 반영한 것으로 제안되어진다. 그리고 1980년대에는 여러 학자들이 최적 실험계획법을 구하는데 시뮬레이티드 어닐링 알고리즘(simulated annealing algorithm)을 이용하였다. Kirkpatrick등(1982)이 메트로폴리스 알고리즘(Metropolis algorithm)에 기초를 둔 어닐링 알고리즘을 처음으로 사용했고, Haines(1987)등에 의하여 활발히 응용되었다. 그리고 강명욱(1995)은 어닐링 알고리즘을 비선형 모형에 적용하였다.

본 논문에서는 Holland(1975)에 의해 개발되어 현재 인공지능 분야에서 많이 이용되고 있는 유전자 알고리즘(genetic algorithm)을 D-최적실험계획법에 처음으로 적용하고자 한다. 유전자 알고리즘은 기존 최적화 알고리즘들과 다음과 같은 차이점이 있다. 유전자 알고리즘은 모수(parameter) 그 자체를 사용하는 것이 아니라, 모수 집합을 코딩해서 사용한다. 유전자 알고리즘은 탐색공간에서 단일해(string)를 사용하는 것이 아니고 해 집단을 사용한다. 유전자 알고리즘은 결정론적인 전이규칙을 사용하는 것이 아닌, 확률적인 규칙을 사용한다. 유전자 알고리즘은 미분 가능성, 연속성 등과 같은 최적화 함수의 정보를 필요치 않고 단지 적합도 함수(fitness)값만을 사용한다.

본 논문은 이러한 장점을 가진 유전자 알고리즘을 이용하여 강명욱(1995)이 제안한 비선형 모형을 선형화하여  $|Z'Z|$ 를 최대화하는 D-최적 알고리즘을 적용하기로 한다.

1장에서는 서론으로 연구의 배경과 목적에 대하여 알아보고, 2장에서는 비선형 모형을 운용하는 방법에 대하여 알아보며, 3장에서는 유전자알고리즘을 최적 실험계획에 적용시키는 문제에 대하여 알아보며, 4장에서는 적용된 유전자 알고리즘에 대한 모의 실험을 하고, 마지막으로 5장에서 결론을 다루기로 한다.

## 2. 비선형 모형에서의 디자인

비선형모형에 대하여 살펴보면, 종속변수  $y$ 와  $q$ 개의 독립변수  $x_1, x_2, \dots, x_q$ 의  $n$ 개의 관측값에 대하여 다음과 같은 비선형모형을 생각한다.

$$Y_i = f(x_i, \theta) + \varepsilon_i, \quad i=1, 2, \dots, n, \quad (2.1)$$

혹은

$$E(Y_i) = f(x_i, \theta) \quad (2.2)$$

$x_i = (x_{i1}, x_{i2}, \dots, x_{iq})'$  이고  $f(X, \theta) = (f(x_1, \theta), f(x_2, \theta), \dots, f(x_n, \theta))'$  라고 하면 (2.1)과 (2.2)는 (2.3)과 같이 표현된다.

$$Y = f(X; \theta) + \varepsilon. \quad (2.3)$$

여기서  $Y$ 는  $1 \times n$ 의 종속변수이고,  $X$ 는  $q \times n$ 의 독립변수이다. 그리고 모수  $\theta$ 는 차수가  $p$ 인 회귀계수벡터이며,  $\varepsilon$ 은 차수가  $n$ 인 오차항 벡터이고 평균  $E(\varepsilon) = 0$ , 분산  $var(\varepsilon) = \sigma^2 I$ 인 정규분포를 따른다고 가정한다. 주어진 비선형모형 (2.3)에서의 함수  $f(X, \theta)$ 를 모수  $\theta$ 의 최소제곱 추정값  $\hat{\theta}$ 에 대하여 Taylor급수 전개를 하고  $\hat{\theta}$ 근방의  $\theta$ 에 대하여 2차 이상 미분한 부분은 무시할 수 있다고 가정하면 비선형모형을 다음과 같이 선형화(linearization)할 수 있다.

$$f(X; \theta) \cong f(X; \hat{\theta}) + g(X) \cdot (\theta - \hat{\theta}) \quad (2.4)$$

여기서  $g(X) = [\partial f(X, \theta) / \partial \theta_i]_{\theta = \hat{\theta}}$ 이고  $g(X)$ 가 선형모형의 독립변수 행렬과 같은 역할을 하기 때문에 식 (2.4)는 선형모형의 특성을 가지므로 앞에서와 같은 선형모형에서 적용한 방법을 그대로 사용할 수 있고,  $g(X)$ 를  $Z$ 라 하면, 최소제곱추정값  $\hat{\theta}$ 의 분산-공분산행렬은  $var(\hat{\theta}) = \sigma^2 (Z'Z)^{-1}$ 이다. 행렬의 크기를 측정하는 일반적인 기준은 행렬식인데,  $var(\hat{\theta})$ 의 행렬식,  $det[var(\hat{\theta})] = det[\sigma^2 (Z'Z)^{-1}]$ 의 값을 최소화하기 위해서는  $det(Z'Z)$ 를 크게 해야한다. 따라서,  $det(Z'Z)$ 를 최대화하는 계획법이 D-최적계획법이며,  $d_{\max} = \max_{x \in X} g(X) (Z'Z)^{-1} g(X)$ 를 최소화하는 계획법이 G-최적계획법이고,  $d_i = \int_{x \in X} g'(X) (Z'Z)^{-1} g(X) d\mu(x)$ 를 최소화하는 계획법이 I-최적 계획법이므로, 이것을 만족하는  $x_{ij} (i=1, 2, \dots, n; j=1, 2, \dots, q)$ 들을 자료로 선택한다.

### 3. 유전자 알고리즘을 이용한 최적디자인

인공 유전 시스템인 유전자 알고리즘은 자연 세계의 진화 현상에 기초한 계산 모델로서 Holland(1975)에 의해 개발된 전역적인 최적화 알고리즘이다. 이것은 모든 생물은 주어진 다양한 환경 속에 적응함으로써 살아남는다는 Darwin의 적자생존(survival of the fittest)의 이론을 기본개념으로 한다. 유전자 알고리즘은 풀고자 하는 문제에 대한 가능한 해들을 정해진 형태의 자료 구조로 표현한 다음 이들을 점차적으로 변형함으로써 점점 더 좋은 해들을 만들어 낸다. 자연계의 모든 생물들은 유전인자인 염색체(chromosome)에 의해서 생식과 유전 등 진화과정을 반복하여 발전해 간다. 유전자 알고리즘에서는 생물학적 유전인자인 염색체에 해당하는 문자열을 가지고 생물과 같은 재생산(reproduction), 교배(crossover), 돌연변이(mutation)를 거쳐서 다음 세대의 새로운 자손(offspring)을 인공적으로 만들어 낸다. 자연의 생물유전을 모방한 연산자들을 반복적으로 적용하여 적합한 해를 탐색한다.

최적화 문제를 위한 진화 프로그램은 실수 표현과 몇 가지 새로운 유전 연산자를 이용한다. 이에 관해 차례로 살펴보자.

#### 3.1 표현

실수 표현을 사용할 때, 각 염색체는 해를 나타낼 수 있는 실수 벡터로 코드화 된다. 각 원소는 초기에 구속조건을 만족하는 영역 내에서 선택되고, 연산자들은 이러한 제약 조건이 계속 만족되도록 고안된다. 이진 표현의 경우에는 이러한 문제가 없고, 연산자의 고안이 꽤 간단하다. 그러나 이러한 측면이 실수 표현의 단점이 되지 않고, 오히려 다음에 언급되어지는 바와 같은 장점을 제공한다. 즉, 실수 표현을 사용하는 경우의 정밀도는 계산 환경에 따라 다르지만, 일반적으로 이진 표현의 경우보다는 훨씬 우수하다. 물론 더 많은 수의 비트를 사용함으로써 이진 표현의 정밀도가 더욱 향상될 수 있기는 하지만, 이 경우에는 계산 속도의 둔화가 뒤따른다. 또한, 실수 표현은 상당히 넓은 영역을 표현할 수 있다. 반면에, 이진 표현은 일단 비트 길이가 정해지면 표현 영역의 확장은 정밀도의 희생 아래에서만 가능해진다.

#### 3.2 초기화 과정과 특수 연산자

초기 개체 집단 중 일부는 적합한 영역(feasible region) 내부로부터 랜덤하게 선택하고, 그 나머지는 해 공간의 경계에 속하는 점을 취하며, 두 그룹 사이의 비율은 연산자들간의 상대적 빈도와 관계가 있게 된다. 또 다른 방법으로는 일부 연산자들은 자손 세대들을 해 공간의 경계 쪽으로 이동시키는 반면 다른 연산자들(전 산술적 교배 등)은 자손 세대들을 적합한 영역의 내부로 분산시켜준다. 여기서는 적합한 영역에서 랜덤하게 선택하는 방법을 취한다.

앞으로 사용할 연산자들은 실수 벡터 공간에서 작용하므로, 이진 표현 방법의 연산자들과는 상당히 다르다. 그러나, 직관적인 유사성 때문에, 이들을 표준 클래스, 즉 교

배와 돌연변이로 구분한다. 그리고, 이들 중 일부는 불균등(non-uniform)이다. 즉, 그 작용이 개체 집단의 나이에 따라 달라지게 된다.

### 3.2.1 돌연변이 그룹

돌연변이는 수렴의 속도를 빨리 이끌어주며, 국부최적에 빠지는 것을 방지하는 유전자 알고리즘의 특징으로, 돌연변이 연산자에서 가장 흔히 사용되는 방법으로는 균등 돌연변이와 불균등 돌연변이가 있으며, 균등 돌연변이는 돌연변이가 모든 구간에서 균등하게 발생하는 것이고, 불균등 돌연변이는 특정한 구간에서 돌연변이가 많이 발생하는 것을 뜻하며 자세한 설명은 다음과 같다.

#### ① 균등 돌연변이

기존의 돌연변이와 유사하게 정의된다. 즉, 염색체

$$x_i^t = \langle v_1, \dots, v_m \rangle \tag{3.1}$$

내의 각 원소  $v_k$ 는 똑같은 돌연변이 확률을 갖는다. 이 연산자가 한번 작용하면, 그 결과로 벡터  $\langle v_1, \dots, v_k', \dots, v_m \rangle$ ,  $1 \leq k \leq m$ 이 생성되는데, 이때  $v_k'$ 는  $k$ 번째 원소가 속하는 영역내의 랜덤 수이다.

#### ② 불균등 돌연변이

지역적 미세 조정을 담당하게 될 연산자 중 하나로써 다음과 같이 정의된다. 염색체  $s_v^t = \langle v_1, \dots, v_m \rangle$ 에 대해서  $k$ 번째 원소  $v_k$ 가 불균등 돌연변이의 대상으로 선택되면 ( $v_k$ 의 정의역은  $[l_k, u_k]$ 임), 이 돌연변이의 결과는 벡터  $s_v^t = \langle v_1, \dots, v_k', \dots, v_m \rangle$ 가 되는데 이 때 변화된 원소  $v_k'$ 는 다음과 같다.

$$v_k' = \begin{cases} v_k + \Delta(t, u_k - v_k) & \text{염색체가 0일 때} \\ v_k - \Delta(t, v_k - l_k) & \text{염색체가 1일 때} \end{cases} \tag{3.2}$$

여기서 함수  $\Delta(t, y)$ 는 영역  $[0, y]$ 내의 값을 제공하되, 세대  $t$ 가 경과할수록 그 값이 0에 가까울 확률이 커지는 성질을 갖는다. 이러한 성질은 이 연산자의 탐색 작업이 초기( $t$ 가 작을 때)에는 전역적으로 균등하게 행해지다가, 점차 지역적이 되도록 해준다. 이 함수의 구체적인 정의는 다음과 같다.

$$\Delta(t, y) = y \cdot (1 - r^{(1 - \frac{t}{T})^\alpha}) \tag{3.3}$$

여기서  $r$ 은  $[0,1]$ 사이의 임의의 수이고,  $T$ 는 최대 세대수, 그리고  $b$ 는 반복 횟수에 대한 종속정도를 결정하는 시스템 매개변수이다.

### 3.2.2 교배 연산자

교배 연산자는 부모의 우수한 형질을 이용하여 자손의 세대에서는 보다 우수한 형질을 유지시키는 연산자로 단순 교배와 산술적 교배가 가장 많이 사용된다. 때때로 두 유전자를 교환하는 취환의 방법이 사용된다.

#### ① 단순교배

가장 단순한 방법으로 주어진 염색체  $x = \langle v_1, \dots, v_m \rangle$ 에 대해 각  $v$ 들 사이의 허용된 지점에서만 교차가 일어날 수 있게 한다.

#### ② 산술적 교배

두 벡터의 1차결합(linear combination)으로 정의된다. 즉,  $s_v^t$ 와  $s_w^t$ 가 교배될 때 그 자손세대는

$$\begin{aligned} s_v^{t+1} &= a \cdot s_w^t + (1-a) \cdot s_v^t \\ s_w^{t+1} &= a \cdot s_v^t + (1-a) \cdot s_w^t \end{aligned} \quad (3.4)$$

가 된다. 이 연산자는  $a$ 값으로 고정된 수를 사용(균등 산술적 교배)하거나, 개체집단의 나이에 따라 변하는 수를 사용(불균등 산술적 교배)할 수 있다. 여기서  $a$ 의 값은 주어진 벡터 ( $s_v$ 와  $s_w$ )에 대해서 연산자가 닫혀 있도록 해주는(즉, 점  $s_v^{t+1}$ 과  $s_w^{t+1}$ 가 볼록한 해 공간  $S$ 에 존재하게 해주는) 가변(dynamic) 매개변수이다. 실제로  $a$ 는 다음 구간에서 랜덤 하게 선택한다.

$$a \in \begin{cases} [\max(\alpha, \beta), \min(\gamma, \delta)], & \text{if } v_k > w_k \\ [0, 0], & \text{if } v_k = w_k \\ [\max(\gamma, \delta), \min(\alpha, \beta)], & \text{if } v_k < w_k \end{cases} \quad (3.5)$$

여기서  $\alpha, \beta, \gamma, \delta$ 는 다음과 같이 정해지는 수이다.

$$\begin{aligned} \alpha &= (l_{(k)}^s - w_k) / (v_k - w_k), \\ \beta &= (u_{(k)}^s - v_k) / (w_k - v_k) \end{aligned}$$

$$\begin{aligned} \gamma &= (l_{(k)}^* - v_k) / (w_k - v_k), \\ \delta &= (u_{(k)}^* - w_k) / (v_k - w_k) \end{aligned} \quad (3.6)$$

$l_{(k)}^*$ 와  $u_{(k)}^*$ 는  $v_k$ 가 취하는 영역의 상한과 하한이며,  $l_{(k)}^*$ 와  $u_{(k)}^*$ 는  $w_k$ 가 취하는 영역의 상한과 하한이다. 이 연산자의 적용 가능성을 높여주기 위해서는 (즉, 이 연산자가 무의해지는  $a=0$ 인 경우의 발생을 막기 위해서는), 두 염색체의 유전인자가 모두 이동 가능한(movable) 위치 중에서  $k$ 를 선택하는 것이 좋다.  $a$ 값은 각 일점 산술적 교배마다, 그리고 각 유전인자마다 독립적으로 결정된다.

본 논문에서 제안하는 모형은 반응표면모형으로, 유전자 프로그램에서 사용할 내부적인 숫자의 표현 방법은 정밀도 문제에서 실수가 이진수보다 우수하기 때문에 실수를 취하기로 하고, 교배연산은 (3.4)의 산술적 교배를 변형하여

$$v_i^{t+1} = \begin{cases} a \cdot v_i^t + (1-a) \cdot v_j^t, & D_c \leq p_c \\ v_i^t & , D_c > p_c \end{cases} \quad (3.7)$$

을 제안한다. 여기서  $v_i^{t+1}$ 은  $t+1$  세대의 임의의 염색체,  $a$ 는 구간  $[0,1]$ 의 난수,  $v_i^t$ 와  $v_j^t$ 는  $t$  세대의 임의의 염색체이다. 단,  $i, j = 1, \dots, N_p$ 이고,  $N_p$ 는 임의로 주어지는 염색체의 수이다. 또한  $D_c$ 는 교배할지 여부를 결정하는 구간  $[0,1]$ 에서 발생되는 난수이고,  $p_c$ 는 사용자가 임의로 주어지는 교배율(crossover rate)이다.

그리고 돌연변이(mutate) 연산자는 (3.2)를 변형한 불균등 돌연변이로

$$v_i' = \begin{cases} v_i + r, & D_m \leq p_m \\ v_i & , D_r > p_m \end{cases} \quad (3.8)$$

을 제안하며, 여기서  $r$ 은 구간  $[-1,1]$ 에서 발생하는 난수를 뜻하고,  $D_m$ 은 돌연변이 연산을 할지 여부를 결정하는 구간  $[0,1]$ 에서 발생하는 난수이며,  $p_m$ 도 사용자가 임의로 주어지는 돌연변이율(mutate rate)이다. 이상을 요약하면 다음과 같은 알고리즘이 완성된다.

- Step 1.  $t \leftarrow 1$ , 목적함수  $P(t)=0$ 으로 초기화한다.
- Step 2.  $[-1,1]$ 에서  $X_{ij}(i=1, \dots, n, j=1, \dots, p)$ 를 난수 발생시킨다.
- Step 3. 목적함수  $|X'X|$ 를 계산한다.

- Step 4. 교배율  $p_c$ 로, 교배 연산으로 (3.7)을 계산한다.  
 Step 5. 돌연변이율  $p_m$ 으로, 돌연변이 연산으로 (3.8)을 계산한다.  
 Step 6. 목적함수  $|X'|$ 를 계산한다.  
 Step 7. 목적함수를 서열화하여 우수한 상위(엘리트 비율)는 보유한다.  
 Step 8. 조건이 맞지 않으면, Step 4로 가고, 조건이 맞으면 인쇄하고 멈춘다.

#### 4. 모의실험

비선형모형으로 자료가 주어진 경우에 보다 효과적인 실험의 결과를 위하여 기존의 자료에 유전자 알고리즘을 이용한 최적실험계획법에 의하여 새로이 추가할 관측값을 선택하는 방법을 모의 실험에 적용한다.

다음에 주어진 모형과 효율성은 강명욱(1995)에서 인용된 예와 값을 사용한다. 촉매제를 투입해서 복합화합물을 단순화합물로 분해하는 화학공정에서 유도된 것이다. 화합물이 분해되는 속도는 여러 가지 요인에 의해서 달라 질 수 있는데 특히 촉매제의 부분압력에 의하여 영향을 받는다고 한다.

$$f(x, \theta) = \frac{\theta_1 \theta_3 (x_2 - x_3 / 1.632)}{1 + \theta_2 x_1 + \theta_3 x_2 + \theta_4 x_3}$$

독립변수  $x_1$ 는 수소의 부분압력,  $x_2$ 는 n화 펜탄의 부분압력이고  $x_3$ 는 이성펜탄의 부분압력(단위 : psia)이다. 종속변수는 1시간에 촉매제 1그램에 대해 생성된 이성펜탄의 무게(단위 : g)를 비율(단위 :  $hr^{-1}$ )로 나타낸 것으로서 기존의 자료는 <표 1>에 수록하였다(Bates와 Watts, 1988: p. 271).

<표 1>에서의 자료를 기초로 하여  $x_1 \in [100, 500]$ ,  $x_2 \in [50, 300]$ ,  $x_3 \in [10, 160]$ 을 각 독립변수의 범위로 사용하였고, 모형에서 사용된 모수의 최소제곱추정값은  $\hat{\theta}_1 = 35.971$ ,  $\hat{\theta}_2 = 0.0708$ ,  $\hat{\theta}_3 = 0.0376$ ,  $\hat{\theta}_4 = 0.1664$ 이다. 유전자 알고리즘을 이용한 최적실험계획법에 의해서 구해진 추가할 관측값은 <표 2>에 제시하였다.

<표 2>에서 보는 바와 같이 추가할 후보점의 수가 증가하면, 유전자 알고리즘을 이용한 방법이 기존의 어닐링 알고리즘을 이용한 것 보다 효율성이 좋은 것으로 나타났다. 강명욱(1995)에서는  $n$ 이 4일 때 목적 함수의 값이  $332.8562 \times 10^6$ 이었으나, 유전자 알고리즘을 이용한 결과에서는  $333.0276 \times 10^6$ 로 효율성이 100.05이고  $n$ 이 5일 때는 강명욱(1995)에서는  $508.0784 \times 10^6$ 이었으나, 본 연구에서는  $611.2343 \times 10^6$ 이고 효율성이 120.30으로 상당히 효과가 뛰어난 것으로 나타났다.



< 표 1 > 화학공정자료

case	$x_1$	$x_2$	$x_3$	$y$	case	$x_1$	$x_2$	$x_3$	$y$
1	205.8	90.9	37.1	3.541	13	297.3	142.2	10.5	5.686
2	404.8	92.9	36.3	2.397	14	314.0	146.7	157.1	1.193
3	209.7	174.9	49.4	6.694	15	305.7	142.0	86.0	2.648
4	401.6	187.2	44.9	4.722	16	300.1	143.7	90.2	3.303
5	224.9	92.7	116.3	0.593	17	305.4	141.1	87.4	3.054
6	402.6	102.2	128.9	0.268	18	305.2	141.5	87.0	3.302
7	212.7	186.9	134.4	2.797	19	300.1	83.0	66.4	1.271
8	406.2	192.6	134.9	2.451	20	106.6	209.6	33.0	11.648
9	133.3	140.8	87.6	3.196	21	417.2	83.9	32.9	2.002
10	470.9	144.2	86.9	2.021	22	251.0	294.4	41.5	9.604
11	300.0	68.3	81.7	0.896	23	250.3	148.0	14.7	7.754
12	301.6	214.6	101.7	5.084	24	145.1	291.0	50.2	11.590

$x_1$  : 수소의 부분압력(psia)       $x_2$  : n화 펜탄의 부분압력(psia)  
 $x_3$  : 이성펜탄의 부분압력(psia)       $y$  : 반응비율(  $hr^{-1}$  )

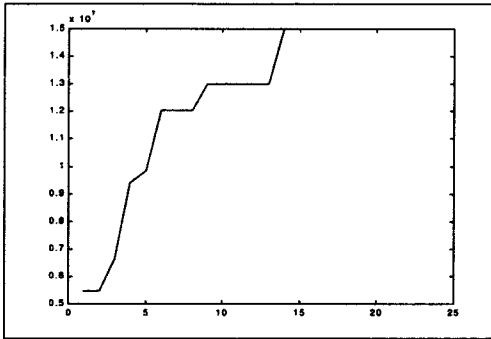
< 표 2 > 비선형 모형의 결과

$n$	추가할 관측값 ( $X_1, X_2, X_3$ )	$ X'X  \times 10^6$	효율성
0		1.8368	100.00
1	(100, 300, 10)	14.9871	100.00
2	(100, 300, 10), (100, 101, 10)	96.2279	100.00
3	(100, 300, 10), (100, 101, 10), (100, 300, 10)	180.5826	100.00
4	(100, 300, 10), (100, 300, 10), (100, 100.75, 10), (100, 101, 10)	333.0276	100.05
5	(100, 300, 10), (100, 300, 93.25), (100, 100.75, 10), (100, 100.75, 10), (100, 300, 10)	611.2343	120.30

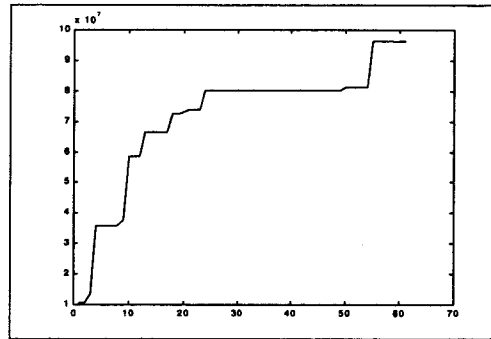
<그림 1>에서 <그림 5>는 최적해에 접근하는 모양을 그래프로 나타낸 것이다. 난수에서 기존 보다 우수한 목적함수를 찾으면, 엘리트 집단으로 저장하고, 그 세대의 최적해로 저장하기 때문에 그래프의 모양이 계단 모양으로 나타난다.

지금까지 예를 이용하여 점검한 바에 따르면 새로운 관측값을 최적실험계획법을 사용하여 추가하면 기존에 있는 자료와 더불어 목적에 맞는 효율적인 자료로 이용할 수

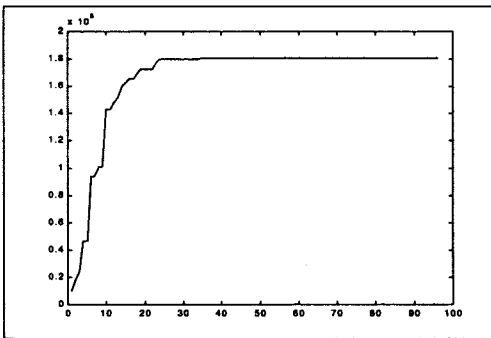
있게 된다. 모의 실험은 Pentium II-233을 그리고 컴파일은 MS Visual C++ Ver. 6.0을 사용하였다.



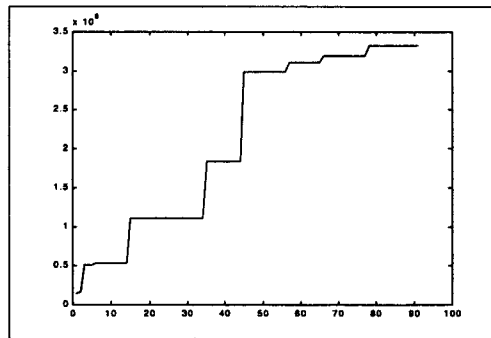
< 그림 1 > 비선형 모형의 n=1 일 때의 그래프



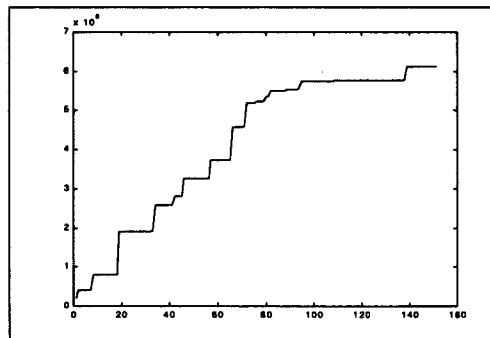
< 그림 2 > 비선형 모형의 n=2 일 때의 그래프



< 그림 3 > 비선형 모형의 n=3 일 때의 그래프



< 그림 4 > 비선형 모형의 n=4 일 때의 그래프



< 그림 5 > 비선형 모형의 n=5 일 때의 그래프

## 5. 결론

본 논문에서는 최적실험계획법의 최적해를 구하는 방법에 처음으로 유전자 알고리즘을 적용하였으며, 효율성에서도 기존의 방법보다 뛰어난 방법이다. 여러 가지 제약 조건이 있는 기존의 방법보다는 유전자 알고리즘을 이용하면, 모의 실험에서 보았듯이 초기값 등의 큰 제약이 없이 국부 최적화에 빠지지 않고, 전역적인 최적값을 찾을 수 있었다.

모의 실험에서 보는 바와 같이 추가할 후보점의 수가 증가하면, 유전자 알고리즘을 이용한 방법이 기존의 어닐링 알고리즘을 이용한 것 보다 효율성이 좋은 것으로 나타났다. 강명욱(1995)에서는  $n$ 이 4일 때 목적 함수의 값이  $332.8562 \times 10^6$ 이었으나, 유전자 알고리즘을 이용한 결과에서는  $333.0276 \times 10^6$ 로 효율성이 100.05이고  $n$ 이 5일 때는 강명욱에서는  $508.0784 \times 10^6$ 이었으나, 본 연구에서는  $611.2343 \times 10^6$ 이고 효율성이 120.30으로 상당히 효과가 뛰어난 것으로 나타났다.

유전자 알고리즘을 사용하면 다른 알고리즘에서는 보기 힘든 몇 가지 좋은 특성들을 갖는다. 첫째, 유전자 프로그램이 최적화할 목적함수가 반드시 연속일 필요는 없다. 일부 최적화 패키지의 경우에는 불연속 목적함수를 전혀 취급할 수 없다. 둘째, 일부 최적화 패키지의 경우에는 “all-or-nothing” 옵션만을 갖는다. 즉, 사용자는 프로그램이 일단 시작되면 종료될 때까지 기다려야 한다. 경우에 따라서는 계산 초반부에 얻어진 근사 해를 얻는 것이 필요할 때가 있는데, 이러한 작업이 불가능한 것이다 반면에 유전자 알고리즘은 사용자에게 이러한 면에서는 상당한 융통성을 주는데, 이는 사용자가 탐색 과정을 관찰하다가 적절한 판단을 내릴 수 있도록 해주기 때문이다. 셋째, 문제 규모의 확대에 따르는 유전자 프로그램의 계산 시간의 증가 비율이 선형적이다. 대부분의 탐색 방법들은 최적화 문제의 규모에 매우 민감하기 마련이다. 또한, 유전자 프로그램은 병렬처리를 통하여 계산 성능을 향상시킬 수도 있다. 다른 최적화 방법들은 이러한 장점을 갖지 못한다.

## 참고문헌

- [1] 강명욱(1995), “비선형모형에서 최적실험계획법의 계산에 관한 연구”, 「품질경영학회지」, 제24권 4호, pp. 59-69.
- [2] Bates, D.M., Watts, D.G.(1988), *Nonlinear Regression Analysis and Its Applications*, John Wiley and Sons, New York.
- [3] Bohachevsky, I.O., Johnson, M.E. and Stein, M.L.(1986), “Generalized Simulated Annealing for Function Optimization,” *Technometrics*, Vol. 28, pp. 209-217.
- [4] Fedorov, V.V.(1969), “Properties and Methods for Construction of Point

- Optimal Designs in Regression Experiments," Preprint No. 5 *LSM*, Izd-vo Moscow State University, Moscow, USSR.
- [5] Fedorov, V.V.(1972), *Theory of Optimal Experiments*, Academic Press: New York.
- [6] Heines, L.M.(1987), "The Application of The Annealing Algorithm to the Construction of Exact Optimal Design for Linear-Regression Models," *Technometrics*, Vol. 29, pp. 439-447.
- [7] Holland, J.H.(1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.
- [8] Kiefer, J.(1971), *The role of Symmetry and Approximation in Exact Design Optimality*. Statistical Decision Theory and Related Topics, 109-118, Academic Press, New York.
- [9] Kirkpatrick, S., Gelatte, C.D., and Vecchi, M.P.(1982), "Optimization by Simulated Annealing," *Science*, Vol. 220, pp. 671-680.
- [10] Lundy, M. and Mees, A.I.(1986), "Convergence of an Annealing Algorithm," *Mathematical Programming*, Vol. 34, pp. 111-124.
- [11] Lundy, M.(1985), "Application of the Annealing Algorithm to Combinatorial Problems In Statistics," *Biometrika*, Vol. 72, pp. 191-198.
- [12] Metropolis, Rosenbluth, Rosenbluth, Teller and Teller(1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, Vol. 21, pp. 1087-1092.
- [13] Michalewicz, Z.(1994), *Genetic Algorithm+Data Structure=Evolution Programs*, 2nd ed., Springer-Verlag, New York.
- [14] Mitchell, T.J.(1974), "An Algorithm for Construction of D-Optimal Experimental Designs," *Technometrics*, Vol. 16, pp. 203-210.
- [15] Mitchell, T.J., and Miller, F.L., JR.(1970), Use of Design Repair to Construct Designs for Special Linear Models. *Math. Div. Ann. Progr. Rept.* (ORNL-4661), Oca Ridge National Laboratory, Oak Ridge, Tennessee, pp. 130-131.
- [16] Wynn, H.P.(1972), "Results in the Theory and Construction of D-Optimum Experimental Designs," *Journal of the Royal Statistical Society*, Ser. B, 34, pp. 133-147.