

# 재시동 조건을 이용한 유전자 알고리즘의 성능향상에 관한 연구

## A Study on Improvement of Genetic Algorithm Operation Using the Restarting Strategy

최 정 목\*      이 진 식\*\*      임 오 강†

Choi, Jung-Muk      Lee, Jin-Sik      Lim, O-Kaung

(논문접수일 : 2001년 9월 1일 ; 심사종료일 : 2002년 3월 26일)

### 요 지

유전자 알고리즘은 적자 생존과 자연선택의 유전이론을 기초로 하여 이루어진 탐색기법이다. 유전자 알고리즘은 미분 정보 등과 같은 부가적인 정보없이 수렴함으로 전역적 최적값을 탐색하는 강인한 탐색기법으로 알려져 있다. 유전자 알고리즘은 연속형의 실제변수를 가지는 문제에서 세대가 계속 진행되어도 목적함수의 개선이 없이 조기에 수렴하는 경우가 있다. 또한 전역적 최적값 근처에서 수렴하지 못하고 목적함수값이 진동하여 수렴속도가 떨어지는 단점이 있다. 본 연구에서는 위와 같은 유전자 알고리즘의 단점을 보완하고자 재시동 조건과 엘리트 보존방법을 제안하였다. 수정된 유전자 알고리즘의 유용성을 검증하기 위해 3부재 트러스와 평면응력 외팔보에 적용하여 수렴 속도의 향상을 확인하였다.

**핵심용어** : 유전자 알고리즘, 재시동 조건, 엘리트 보존방법

### Abstract

The genetic algorithm(GA), an optimization technique based on the theory of natural selection, has proven to be relatively robust means to search for global optimum. It is converged near to the global optimum point without auxiliary information such as differentiation of function. When studying some optimization problems with continuous variables, it was found that premature saturation was reached that is no further improvement in the object function could be found over a set of iterations. Also, the general GA oscillates in the region of the near global optimum point so that the speed of convergence is decreased. This paper is to propose the concept of restarting and elitist preserving strategy as a measure to overcome this difficulty. Some benchmark examples are studied involving 3-bar truss and cantilever beam with plane stress elements. The modifications to GA improve the speed of convergence.

**Keywords** : Genetic algorithm, Restarting strategy, Elitist preserving strategy

† 책임저자, 정회원 · 부산대학교 기계공학부 교수 iDOT연구원  
전화번호: 051-510-2306; FAX: 051-582-9164  
E-mail: oklim@pusan.ac.kr

\* 부산대학교 기계설계공학과 대학원

\*\* 경기공업대학 컴퓨터 응용설계계열 치공구 설계전공 전임강사

• 이 논문에 대한 토론을 2002년 9월 30일까지 본 학회에  
보내주시면 2002년 9월호에 그 결과를 게재하겠습니다.

## 1. 서 론

최적설계 문제에는 순차이차계획법(sequential quadratic programming, SQP), 점근선이동법(the method of moving asymptotes, MMA), 일반화된 환산경사법(generalized reduced gradient method, GRG) 등과 같은 수학적인 비선형 계획 알고리즘<sup>1)</sup>이 많이 이용되고 있다. 이런 비선형 계획 알고리즘은 탐색방향을 정하기 위하여 기울기 정보, 즉 목적함수와 제약조건에 대한 설계변수의 일차 미분정보가 필요하다. 따라서 설계공간이 불연속이거나 설계변수 및 구속조건이 많으며, 설계변수의 형태가 이산형 또는 정수형인 경우에는 민감도해석이 어려울 뿐 아니라 다량의 설계해가 존재할 가능성이 있다. 따라서, 효율적 탐색을 위해서는 확률적 탐색기법에 근거한 방법들이 효과적이며, 유전자 알고리즘(genetic algorithm, GA) 또는 냉각모사기법 등이 대표적으로 쓰이고 있는데, 특히 유전자 알고리즘은 복합재료 구조물 등을 비롯한 다양한 구조최적화 분야에 적용되고 있다.<sup>2)</sup> 그 예로 Coello<sup>3)</sup>와 황선일<sup>4)</sup> 등은 여러 트러스 구조물에 대한 치수최적설계를 행하였으며, Daly<sup>5)</sup>는 비행기의 수평꼬리날개 리브(rib) 부분의 형상최적설계를 수행하였다.

유전자 알고리즘은 Holland에 의하여 종합·체계화된 알고리즘으로 군집(population)이라는 설계점들의 집합을 이용하여 개연적인 변이법칙들에 의해 세대와 세대가 진행되면서 최적점을 찾아간다.<sup>6)</sup> 군집이라는 설계점들의 집합을 이용하여 탐색하기 때문에 전역적 최적점을 찾기가 용이하며, 탐색하기 위한 정보로서 목적함수 값만을 이용하기 때문에 미분가능일 필요가 없고, 다른 부수적인 정보도 필요로 하지 않는다. 그렇지만, 유전자 알고리즘에서는 군집을 이용하기 때문에 최적점에 도달하기 위하여 많은 함수계산이 요구되어서 수학적인 비선형 계획 알고리즘에 비하여 수렴속도가 느리며, 국부탐색능력이 떨어져서 최적점 근처에서 수렴하지 못하고 함수값이 진동하여 수렴속도가 떨어지는 단점이 있다.<sup>7)</sup> 위와 같은 유전자 알고리즘의 단점을 개선하기 위해 Jenkins<sup>8)</sup>는 교배(crossover) 과정을 수정하였으며 교배율과 돌연변이율을 세대마다 변화시키는 방법을 제안하였고 Mahfouz<sup>9)</sup> 등은 엘리트 전략을 제안하여 유전자 알고리즘을 강철구조물의 최적설계에 이용하였다. Tahk<sup>10)</sup>과 Barbosa<sup>11)</sup>는 공진화(Coevolutionary)를 이용하여 제약조건을 효율적으로 다루려고 하였다. 또한 조영석<sup>12)</sup>은 순위 벌칙 방법과 기존의 민감도를 이용하는 최적화 방법과의 적절한 결합을 통하여 유전자 알고리즘에서 제약조건을 효율적으로 처리하고자 하였다. 본 연구에서는 유전자 알고리즘의 위와 같은 단점을 개선하고자 세대의 진행에 따라 선택되는 설계점들을 한 세대전의 세대와 현재 세대를 비교하여 적합성이 좋은 배열을 선정하는 방법과 재시동 조건을 이용하여 설계변수의 설계영역을 변화시키는 방법을 제안한다. 몇 가지 수치예제를 통하여 단순 유전자 알고리즘(basic genetic algorithm)으로 최적설계를 수행한 결과와 수정을 한 유전자 알고리즘(modified genetic algorithm, MGA)의 결과를 비교하여 수렴율과 수렴속도를 비교하였다.

로 다루려고 하였다. 또한 조영석<sup>12)</sup>은 순위 벌칙 방법과 기존의 민감도를 이용하는 최적화 방법과의 적절한 결합을 통하여 유전자 알고리즘에서 제약조건을 효율적으로 처리하고자 하였다. 본 연구에서는 유전자 알고리즘의 위와 같은 단점을 개선하고자 세대의 진행에 따라 선택되는 설계점들을 한 세대전의 세대와 현재 세대를 비교하여 적합성이 좋은 배열을 선정하는 방법과 재시동 조건을 이용하여 설계변수의 설계영역을 변화시키는 방법을 제안한다. 몇 가지 수치예제를 통하여 단순 유전자 알고리즘(basic genetic algorithm)으로 최적설계를 수행한 결과와 수정을 한 유전자 알고리즘(modified genetic algorithm, MGA)의 결과를 비교하여 수렴율과 수렴속도를 비교하였다.

## 2. 유전자 알고리즘

유전자 알고리즘은 적자 생존과 자연선택의 유전이론을 기초로 하여 이루어진 탐색방법이다. 즉, 자연 생태계에서 주어진 환경에 적합한 개체일수록 여러 세대가 지나도 살아남을 수 있는 확률이 커지고, 열성형질일수록 더욱 쇠퇴한다는 유전이론에 기초를 둔 알고리즘이다.

### 2.1 유전자 알고리즘의 연산과정

유전자 알고리즘에서는 진화 연산자(evolution operation)와 유전 연산자(genetic operation)가 있다. 진화 연산자에는 선택(selection)이 있으며, 유전 연산자에는 교배(crossover)와 돌연변이(mutation)가 있다. 유전자 알고리즘은 이 연산자들의 반복과정에 의하여 최적값을 탐색하게 된다.

#### 2.1.1 선택(selection)

선택은 적합도가 좋은 해들이 적합도가 떨어지는 해들에 비하여 잘 보존될 수 있도록 한다. 즉, 주어진 세대의 군집에서 적합도가 높은 배열일수록 다음 세대에 선택될 확률이 높아진다. 일반적인 선택방법으로는 룰렛 휠 선택법(Roulette wheel selection), 토너먼트 선택법(Tournament selection), 순위 선택법(Ranking selection), 엘리트 보존 선택법(Elitist preserving selection) 등이 있다. 일반적인 유전자 알고리즘에서는 룰렛 휠 선택법과 토너먼트 선택법이 많이 사용된다.

2.1.2 교배(crossover)

교배는 현 세대의 배열을 조합하여 새로운 배열을 생성하는 역할을 한다. Fig. 1과 같이 배열의 길이가  $k$  인 배열에서 교배위치를 임의로 정하고 그 위치를 기준으로 서로 짝이 되는 배열의 부분과 교환이 일어난다.

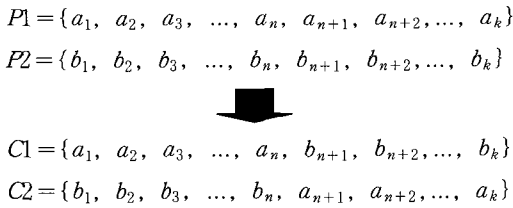


Fig. 1 Example of crossover process

일반적으로 유전자 알고리즘에서의 교배과정을 Fig. 2에 도시하였다. A의 경우가 가장 일반적인 방법인 전체 배열의 한 곳을 교배위치로 정하는 것이다. 그러나 A의 경우 배열의 길이가 길어지면 한 곳을 기준으로 교배가 일어나기 때문에 설계변수의 변화가 적어지게 되고 그 결과 알고리즘의 속도를 떨어지게 만드는 단점이 있다. 이런 단점을 보완하기 위하여 교배위치를 두 곳, 세 곳으로 정할 수 있으며, 또한 B와 C처럼 설계변수를 나타내는 각각의 부분배열의 임의의 한 곳을 교배위치로 정할 수 있다. B는 모든 설계변수를 나타내는 부분배열의 교배가 한 쌍의 배열에서 일어나고, C의 경우는 각각의 부분배열의 교배가 일어나는 쌍이 바뀐다는 차이가 있다. 본 연구에서는 배열의 변화를 다른 방법보다 많이 줄 수 있는 C의 방법으로 교배를 수행하였다.

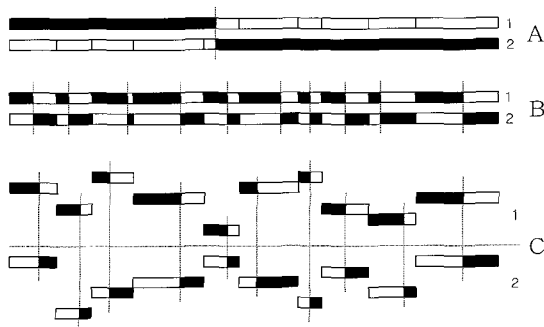


Fig. 2 Alternative forms of crossover

2.1.3 돌연변이(mutation)

돌연변이는 군집이 가지는 탐색공간의 제한성을 극복하기 위하여 배열을 전혀 다른 형태로 변화시키는 역할을 한다.

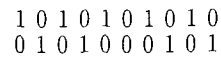


Fig. 3 Representation of one pair string

Fig. 3와 같은 경우 두 개체가 서로 교배를 할 경우 어느 곳에서 교배가 일어나더라도 6번째 자리에 1이 올 수 없다. 같은 맥락에서 만약 충분한 배열의 수를 사용하여 충분한 진화를 수행하여 대부분의 값이 비슷한 구조로 이루어졌다고 한다면 더 이상 교배만으로는 새로운 값을 찾을 수 없게 된다. 또, 세대가 진행됨에 따라 설계점들이 너무 한 곳으로 치우칠 수 있기 때문에 돌연변이를 이용하여 이런 현상을 방지할 수 있다.

2.2 적합도 함수(Fitness function)

처음 유전자 알고리즘은 비제한조건 문제를 다루기 위해서 만들어 졌기 때문에, 제한조건을 처리하기 위한 여러 가지 방법들이 연구되어져 왔다. 그 중 하나가 진화과정 중 발생한 불용영역의 해는 군집에서 무조건 버리는 방법이다. 그러나 이 방법은 불용영역의 해가 너무 자주 발생하면 유전자 알고리즘은 제한조건을 만족시키는 몇 개의 해를 찾기 위하여 많은 시간을 낭비하게 되고, 실제의 전역 최적값에 가까이 있는 해일지라도 제한조건을 위배하면 무조건 버리기 때문에 알고리즘의 탐색과정을 위해서는 바람직 하지 않다. 이에 대부분의 경우 유전자 알고리즘은 제한조건을 처리하기 위하여 벌칙함수 방법을 사용한다.

벌칙함수는 제약 최적화문제를 비제약 최적화문제로 전환하기 위하여 이용되는 것으로 불용영역의 해에 의하여 구해진 목적함수 값에 제약조건 위배량에 대한 벌칙량을 부과하여 표현한다. 최적화 문제가 다음과 같이 표현될 때,

$$\begin{aligned} &\text{maximize } f(x) && (1) \\ &\text{subject to. } g_i(x) \leq b_i, \quad i=1,2,\dots,m && (2) \end{aligned}$$

Yokoda와 Gen에 의하여 제시된 벌칙함수<sup>13)</sup>는 다음과 같다.

$$fit(x) = f(x)p(x) \quad (3)$$

$$\text{여기서, } p(x) = 1 - \frac{1}{m} \sum_{i=1}^m \left( \frac{\Delta b_i(x)}{b_i} \right)^a \quad (4)$$

$$\Delta b_i(x) = \max \{0, g_i(x) - b_i\} \quad (5)$$

여기서,  $fit(x)$ 는 적합도함수,  $p(x)$ 는 벌칙함수,  $\Delta b_i(x)$ 는  $i$ 번째 제약조건의 위배량이며,  $a$ 는 벌칙함수의 가중치를 조정하는 상수로서 문제의 성격에 따라 바뀌는 값이다. Yokoda가 제시한 벌칙함수는 실행결과 설계영역이 넓은 문제에서 벌칙함수가 적합도 함수에 미치는 영향이 작아지는 경향이 있어 불용영역의 해를 너무 많이 군집속에 남게 하는 단점이 있었다. 또한 문제의 성격에 따라 변수들의 조정이 필요하여 사용자의 불편이 따른다. 따라서 본 연구에서는 Yokoda가 제시한 벌칙함수를 다음과 같이 변환하여 적합도 함수에서 벌칙함수의 영향력을 높여 주었다.

$$fit(x) = f(x) - p(x) \quad (6)$$

$$\text{여기서, } p(x) = f_{\max} \times \frac{1}{m} \sum_{i=1}^m \left( \frac{\Delta b_i(x)}{b_i} \right) \quad (7)$$

여기서,  $f_{\max}$ 는 군집 중 가장 큰  $f(x)$ 의 값이다. 벌칙함수에서  $f_{\max}$ 의 영향이 크게 되므로 벌칙함수의 목적함수에 대한 영향력이 커지게 된다. 따라서, 불용영역의 해에 대하여 높은 벌칙값을 가하여 유용영역의 해보다 선택될 확률이 줄어들어 많은 유용영역의 설계해들이 존재하게 되어 빠른 수렴경향을 보여준다.

### 3. 수정된 유전자 알고리즘

최적설계문제에서 수렴속도를 개선시키기 위해서는 설계점들이 전영역 최적점 근처까지 빠르게 접근하고, 국부탐색능력을 향상시켜야 한다. 따라서 본 논문에서는 유전자 알고리즘의 현 세대의 좋은 적합도를 가진 열들에 의해 생성된 다음 세대의 배열들은 현 세대의 나쁜 적합도를 가진 배열들에 의해 생성된 다음 세대의 배열들에 비해 좋은 적합도를 가질 확률이 높다는 기본 가정을 따라서 유전자 알고리즘의 성능을 향상시키고자 하였다.

### 3.1 엘리트 보존 방법

군집속에 좋은 적합도를 가지는 배열들이 많이 포함되어 있으면 교배나 돌연변이 과정을 거치면서 다음 세대에 보다 좋은 적합도를 가지는 배열이 생성될 확률이 높아진다. 군집속에 좋은 적합도를 가지는 배열을 많이 포함시키기 위하여 다음과 같은 단계를 거쳤다.

- 단계 1. 알고리즘을 실행하기 이전에 예비로 알고리즘에 사용될 군집의 크기보다 몇 배 더 큰 군집을 생성하여 그 중에서 적합도가 좋은 배열을 선택하고 초기 군집을 생성하여 알고리즘을 실행한다.
- 단계 2. 현 세대와 한 세대 전의 세대에서 저장된 배열들과 그 적합성을 비교하여 적합도가 좋은 순서대로 배열들을 선택하여 새로운 군집을 형성한다.
- 단계 3. 현 세대까지 가장 좋은 적합도를 가지는 배열은 교배나 돌연변이 과정을 거치지 않고 다음 세대에 보존한다.

단계 1에서 엘리트 보존 방법을 사용한 유전자 알고리즘은 초기에 단순 유전자 알고리즘보다 몇 배 많은 계산속도를 가진다. 하지만, 군집속에 좋은 적합도를 가지는 배열들이 많이 포함되므로 수렴이 빨리 될 확률이 높아지며 결과적으로 적은 계산과정을 가질 수 있다. 단계 3에서는 교배나 돌연변이에 의하여 배열이 파괴되지 않으므로 엘리트 개체의 유전자가 군집 안에 급속히 확산될 가능성이 높기 때문에 국소해에 빠질 위험이 있다. 따라서 선택방법으로 룰렛 휠 선택법을 사용하여 그 단점을 보완하고자 하였다.

### 3.2 제시동 조건

일반적인 유전자 알고리즘은 수렴을 판정하는 기준이 없기 때문에 충분한 최대의 세대를 입력값으로 주어 그 세대 동안에 함수값이 개선되지 않으면 수렴된 것으로 간주한다. 따라서 국부탐색능력이 떨어지는 유전자 알고리즘은 최적점 근처에서 함수값의 개선이 없어 조기에 수렴하거나 설계영역이 넓은 경우는 개선되는 속도가 떨어지는 단점이 있다. 이 단점을 극복하기 위하여 경우의 수를 늘릴 수 있도록 군집의 크기를 증가시키거나 설계영역을 줄이는 방법이 있다. 본 연구에서는

재시동 조건이라는 알고리즘을 정지하는 조건을 제안하여 알고리즘의 설계영역을 점차로 줄이는 방법을 제안한다. 재시동이 일어나기 위해서는 다음과 같은 재시동 조건을 만족하여야 한다.

$$|fitness_i - fitness_{i-1}| \leq \beta, \quad i = 2, 3, \dots, generation \quad (8)$$

여기서  $fitness_i$ 는  $i$ 번째 세대의 적합도 함수 중 가장 적합도가 좋은 함수이다.  $\beta$ 는 문제의 성격에 맞게 바뀌는 크기가 작은 상수이다. 식(8)과 같은 재시동 조건을 만족하면 알고리즘은 새로운 설계영역을 가지고 다시 시작하게 된다. 새로운 설계영역은 다음과 같이 구해진다.

$$X_i^L \leq X_i \leq X_i^U, \quad i = 1, 2, \dots, ND \quad (9)$$

여기서,  $X_i^L$ 와  $X_i^U$ 는 각각  $i$ 번째 설계변수의 초기 하한값과 상한값이며,  $ND$ 는 설계변수의 개수이다.

$$A_i = \frac{X_i^U}{X_i^L} \quad (10)$$

식(10)은 설계변수의 상한과 하한의 차수차이를 구하기 위한 식이다. 설계변수 영역의 변화량을 결정하기 위하여 만약  $A_i$ 가  $10^j$ 로 표현되는 값보다 작으면 그때의  $j$ 값을 선택하여 다음과 같은 식으로 설계변수의 변화량을 구한다.

$$offset_i = n \times j \quad (11)$$

여기서,  $offset_i$ 는 설계변수  $i$ 의 영역 변화량이며,  $n$ 은 설계변수 영역의 변화량을 결정하는 상수이며 문제의 성격에 따라 변한다.  $n$ 의 값이 너무 작으면 설계영역이 갑자기 줄어드는 경향이 발생하여 국소해에 빠질 염려가 있으므로 보통 0.2~0.5 정도의 값을 사용한다. 설계변수의 새로운 영역은 다음과 같이 구해진다.

$$p_{min} = \max(X_i^L, (1 - offset_i) \times X_i) \quad (12)$$

$$p_{max} = \min(X_i^U, (1 + offset_i) \times X_i) \quad (13)$$

$$p_{min} \leq X_i^k \leq p_{max}, \quad k = 1, 2, \dots, NR \quad (14)$$

여기서,  $NR$ 은 재시동 조건이 발생한 개수이다.

### 3.3 교배율과 돌연변이율

유전자 알고리즘의 성능은 내부 파라미터에 영향을 받으며, 그 중 교배율과 돌연변이율은 중요한 파라미터이다. 기존의 유전자 알고리즘은 초기에 설계자의 직관에 의하여 임의의 고정된 값을 주었으나, 제안한 방법은 군집의 배열들의 분포에 의하여 교배율과 돌연변이율을 변화시키고자 하였다. 매 세대마다 모집단 내의 적합도 함수값들의 분산값을 구해서 이전 세대의 분산값보다 크면 다양성의 관점에서 우수하다고 보아 돌연변이율을 낮추고 교배율을 높여 그만큼 임의의 영역을 찾는 확률을 줄여주고, 이전 세대의 분산값보다 작으면 다양성의 관점에서 부족하다고 보아 돌연변이율을 높이고 교배율을 낮춰 임의의 영역을 찾는 확률을 늘려준다. 분산값의 변화에 따른 교배율과 돌연변이율의 변화는 Table 1에 나타내었다.

## 4. 수치해석

본 연구에서 제안한 수정된 유전자 알고리즘을 이용한 최적설계 문제로 선형·비선형 제한조건을 가지고 있는 최적화 문제, 여러 트러스 부재문제와 평면응력요소로 이루어진 외팔보 문제를 선정하여 수정된 유전자 알고리즘을 이용하여 최적설계를 수행하고, 그 결과들을 단순 유전자 알고리즘의 수행결과와 비교하였다.

### 4.1 3부재 트러스

대칭인 3부재 트러스를 Fig. 4에 도시하였고 설계 데이터는 Table 2에 제시하였다. 본 설계에 사용한 요소는 절점당 자유도가 2인 트러스 요소를 사용하였다. 사용한 요소개수는 3개, 절점수는 4개이다. 설계변수는 각 부재의 단면적으로 요소 1과 3은 대칭으로 가정하여 설계변수를 2개로 선정하였다. 목적함수는 부재의 총무게 최소화이고 제한조건은 각 부재에 작용하는 응력  $\sigma_i$ 가 허용응력  $\sigma_a$ 를 초과하지 않도록 하며, 압축 응력을 받는 부재는 좌굴되지 않도록 하였다. 응력은 유한요소법에 의해 해석하였다.

Table 1 Change of probability of mutation and crossover

$\sigma_{pre} > \sigma_{cur}$			$\sigma_{pre} < \sigma_{cur}$		
$\frac{\sigma_{pre} - \sigma_{cur}}{\sigma_{pre}} = m$	$P_m$	$P_c$	$\frac{\sigma_{cur} - \sigma_{pre}}{\sigma_{cur}} = m$	$P_m$	$P_c$
$m \leq 0.1$	0.0275	0.225	$m \leq 0.1$	0.0225	0.275
$0.1 < m \leq 0.2$	0.03	0.2	$0.1 < m \leq 0.2$	0.02	0.3
$0.2 < m \leq 0.3$	0.0325	0.175	$0.2 < m \leq 0.3$	0.0175	0.325
$0.3 < m \leq 0.4$	0.035	0.15	$0.3 < m \leq 0.4$	0.015	0.35
$0.4 < m \leq 0.5$	0.0375	0.125	$0.4 < m \leq 0.5$	0.0125	0.375
$0.5 < m \leq 0.6$	0.04	0.1	$0.5 < m \leq 0.6$	0.01	0.4
$0.6 < m \leq 0.7$	0.0425	0.075	$0.6 < m \leq 0.7$	0.0075	0.425
$0.7 < m \leq 0.8$	0.045	0.05	$0.7 < m \leq 0.8$	0.005	0.45
$0.8 < m \leq 0.9$	0.0475	0.025	$0.8 < m \leq 0.9$	0.0025	0.475
$0.9 < m$	0.05	0.0	$0.9 < m$	0.0	0.5

$\sigma_{pre}$  : distribution of the previous population  
 $P_m$  : mutation probability

$\sigma_{cur}$  : distribution of the current population  
 $P_c$  : crossover probability

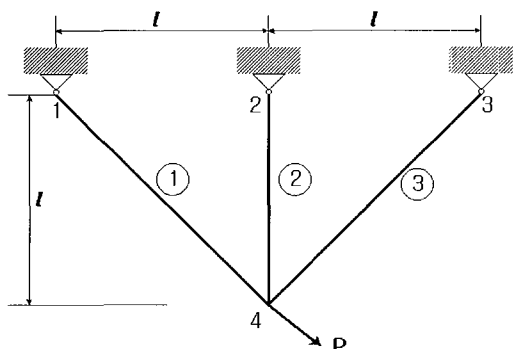


Fig. 4 Three-bar truss

Table 2 Design data for three bar truss

Design data	value
Weight density, $\gamma$ (N/mm <sup>3</sup> )	$27.14 \times 10^{-6}$
Length, $l$ (mm)	2540
Modulus of elasticity, $E$ (kN/mm <sup>2</sup> )	68.95
Load, $P$ (kN)	88.96
Allowable tensile stress, $\sigma_a$ (kN/mm <sup>2</sup> )	0.1379
Allowable compressive stress, $\sigma_c$ (kN/mm <sup>2</sup> )	0.1034
Max generation number	500

단순 유전자 알고리즘(Pure-GA)와 수정된 유전자 알고리즘(MGA)에서 초기 교배율 0.5, 초기 돌연변이율 0.025로 하여 최적설계를 수행하였다. 또한, MGA-2는 식(8)에 의해서 재시동이 일어나며, 재시동 단계를

3번(1st,2nd,3rd step) 반복한후 최종단계(Final step)에서 수렴하는 것으로 하였다. Table 3에 3부재 트러스의 최적설계 결과가 나타나 있다. 재시동 조건을 사용한 MGA-2는 초기의 설계변수 영역이 0.001부터 100000사이의 값이 었으나, 재시동이 3번 반복한 후에는 각 설계변수의 범위가 0.001부터 2983.05과 16.442로 줄어든 것을 알수 있다. 단순 유전자 알고리즘(Pure-GA)는 최대 세대수인 500세대가 지나도록 수렴하지 못하였고 수정된 유전자 알고리즘(MGA)의 경우는 30세대 이전에 모두 수렴하였다. 설계변수가 2개로 설계변수가 작은 문제로 군집의 크기를 100으로 한 Pure-GA의 수렴정도가 군집의 크기를 20으로 한 MGA-1의 경우보다 좋다. 설계변수가 작은 문제에서 군집의 크기가 큰 경우 많은 설계점들이 교배나 돌연변이 과정을 거치면서 다양한 설계점을 생성하기 때문에 설계영역이 넓은 경우에도 설계해들의 적합도의 향상을 보여 수렴정도가 좋은 것으로 사료된다. 그러나 수렴속도가 느리고 수렴하지 못한 결과를 볼 수 있다. 엘리트 보존 방법과 재시동 조건을 사용한 경우 설계영역이 넓은 문제에서 잘 적응됨을 MGA-2의 결과에서 볼 수 있다.

#### 4.2 평면응력요소로 구성된 외팔보

평면응력요소로 구성된 외팔보 문제는 길이가 500mm,

Table 3 Optimization results for three-bar truss

Method		Pure-GA	MGA-1 <sup>(1)</sup>		MGA-2 <sup>(2)</sup>			
Population size		100	20	100	20			
Restarting Condition <sup>(3)</sup> (the value of $\beta$ )		.	.	.	1st step ( $1 \times 10^{-4}$ )	2nd step ( $1 \times 10^{-5}$ )	3rd step ( $1 \times 10^{-5}$ )	Final step
Min. design value( $X_i^L$ )	1	0.001	0.001	0.001	0.001	0.001	0.001	0.001
	2	0.001	0.001	0.001	0.001	0.001	0.001	0.001
Max. design value( $X_i^U$ )	1	100000	100000	100000	100000	19917.43	11890.76	2983.05
	2	100000	100000	100000	100000	24934.24	235.70	16.442
Optimum design value	1	1562.50	1599.12	781.35	4742.25	2831.14	710.25	710.25
	2	42.535	28.611	0.001	5936.72	56.120	4.836	4.836
Obtect function(N)		307.59	313.77	152.35	1333.89	555.88	138.82	138.82
Random number generation		2279167	41977	255230	6023	26228	29309	40770
FEM iteration		50000	960	5800	160	680	800	1240
Generation		500	23	28	3	15	17	27

- (1) MGA-1: elitist preserving strategy
- (2) MGA-2: elitist preserving strategy with restarting strategy
- (3) If the change of fitness function value less than restarting condition  $\beta(1 \times 10^{-5})$ , then generation of GA will be terminated in pure-GA, MGA-1, and Final step of MGA-2

높이가 100mm인 외팔보를 절점당 자유도가 2인 평면응력요소를 이용하여 유한요소 모델을 만들고 최적설계를 수행하였다. 절점의 개수는 12개, 요소수는 5이다. 설계변수는 각 부재의 두께로 하여 설계변수를 5개로 선정하였다. 목적함수로는 부재의 총무게의 최소화를 선정하였고 제한조건은 각 부재에 작용하는 응력  $\sigma_i$ 이 허용응력  $\sigma_a$ 를 초과하지 않도록 하며 각 절점의 변위  $\delta_j$ 가 허용변위  $\delta_a$ 를 초과하지 않도록 하였다. Fig. 5에 평면응력요소로 구성된 외팔보를 도시하였다.

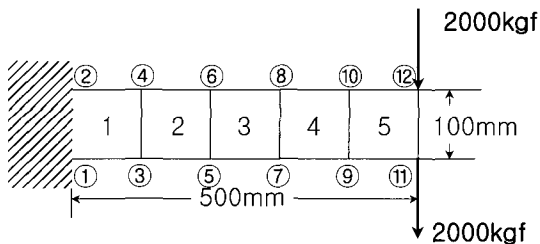


Fig. 5 Cantilever beam with quadrilateral elements

본 설계에 사용된 입력값은 Table 4와 같다.

단순 유전자 알고리즘(Pure-GA)와 수정된 유전자 알고리즘(MGA)에서 초기 교배율 0.5, 초기 돌연변이율 0.025로 하여 최적설계를 수행하였다. Table 5에 평면응력요소로 구성된 외팔보의 최적설계 결과가 나

Table 4 Design data for the cantilever beam problem with plane stress elements

Design data	value
Weight density $\gamma(\text{kg} \cdot \text{s}^2/\text{mm}^4)$	$73.73 \times 10^{-7}$
Modulus of elasticity $E(\text{kgf}/\text{mm}^2)$	21100
Allowable stress $\sigma_a(\text{kgf}/\text{mm}^2)$	15.0
Allowable displacement $U_a(\text{mm})$	3.0
Poisson's ratio $\nu$	0.3
Max generation number	500

타나 있다. MGA-2에서 재시동 단계마다 설계변수의 영역이 줄어들고 있다. 단순유전자 알고리즘은 500세대가 지나도록 수렴하지 못하였으며, 수정된 유전자 알고리즘의 결과는 재시동 조건을 이용한 MGA-2가 MGA-1보다 좋은 결과를 나타냄을 볼 수 있다.

### 5. 결 론

본 연구는 유전자 알고리즘의 수렴성을 개선하는 방안으로 엘리트 보존 방법과 재시동 조건을 제안하였다. 본 연구에서 제안한 수정된 유전자 알고리즘을 3부재 트러스 구조물과 평면응력요소로 구성된 외팔보에 각각 적용하여 그 유용성을 검증하였으며, 단순유전자 알고리즘을 이용한 경우와 비교하였다. 그 결과, 설계영

Table 5 Optimization results for cantilever beam problem with plane stress elements

Method		Pure-GA	MGA-1 <sup>(1)</sup>		MGA-2 <sup>(2)</sup>			
Population size		100	20	100	20			
Restarting condition <sup>(3)</sup> (the value of $\beta$ )		.	.	.	Initial ( $1 \times 10^{-6}$ )	1st step ( $1 \times 10^{-6}$ )	2nd step ( $1 \times 10^{-6}$ )	Final step
Min. design value ( $X_i^L$ )	1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	2	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	3	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	4	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	5	0.1	0.1	0.1	0.1	0.1	0.1	0.1
Max. design value ( $X_i^U$ )	1	1000	1000	1000	1000	165.63	165.62	165.82
	2	1000	1000	1000	1000	125.03	124.02	124.70
	3	1000	1000	1000	1000	76.491	76.491	76.655
	4	1000	1000	1000	1000	49.245	49.245	49.293
	5	1000	1000	1000	1000	25.990	25.990	22.304
Optimum design value	1	62.596	63.703	62.597	63.703	63.703	63.778	62.279
	2	106.73	48.071	62.608	48.087	48.087	47.963	47.932
	3	85.134	34.774	35.255	34.774	34.774	34.843	34.849
	4	169.28	22.389	31.349	22.389	22.389	22.406	22.404
	5	23.068	11.819	15.726	11.819	11.819	10.138	9.950
Obtect function(N)		32.943	13.328	15.302	13.328	13.328	13.207	13.081
Random number generation		6612574	355075	1379852	302372	309693	335519	400553
FEM iteration		50000	2720	10600	2320	2440	2760	3521
Generation		500	67	52	57	59	66	84

(1) MGA-1: elitist preserving strategy

(2) MGA-2: elitist preserving strategy with restarting strategy

(3) If the change of fitness function value less than restarting condition  $\beta (1 \times 10^{-6})$ , then generation of GA will be terminated in pure-GA, MGA-1, and Final step of MGA-2

역이 넓은 경우, 단순 유전자 알고리즘은 국부탐색능력이 떨어짐으로 인한 초기 수렴과 많은 군집을 요구하는 단점이 있었으나, 제안한 방법은 작은 군집으로서, 초기의 넓은 설계영역을 점차 줄이므로써, 함수계산량과 수렴성을 개선하는 결과를 볼수 있었다.

### 감사의 글

본 연구는 한국과학기술재단 지정 최적설계신기술 연구센터의 지원으로 수행되었습니다. 재단의 지원에 감사드립니다.

### 참 고 문 헌

1. M. R. Ghasemi and E. Hinton, "Truss Optimization Using Genetic Algorithms", *CIVIL-COMP Ltd*, 1996, pp.59~75
2. 이종수, "유전알고리즘의 최적설계 적용", 기계저널, 40권, 제3호, 2000, pp.38~42
3. C. A. Coello and A. D. Christiansen, "Multi-objective Optimization of Trusses Using Genetic Algorithms", *Computer & Structures*, Vol. 75, No. 6, 2000, pp.647~660
4. 황선일, 조홍동, 한상훈, "유전알고리즘을 이용한 트러스 구조물의 이산 최적설계", 한국 전산 구조 공학회 논문집, 제14권 제2호, 2001, pp.97~106
5. P. Daly, "Multidisciplinary Structural Optimization Utilizing Genetic Algorithm Capabilities", *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization - Part 1*, 1998, pp.553~558
6. R. T. Haftka and Z. Gürdal, *Elements of Structural Optimization*, Kluwer Academic Publishers, Boston, 1992, pp.481



7. 배충규, "수렴율이 개선된 유전이론을 통한 구조물 최적설계", 석사학위논문, 부산대학교, 1996, pp.38
8. W. M. Jenkins, "On the Application of Natural Algorithms to Structural Design Optimization", *Engineering Structures*, Vol. 19, No. 4, 1997, pp.302~308
9. S. Y. Mahfouz, V. V. Toropov and R. K. Westbrook, "Improvements in the Performance of a Genetic Algorithm : Application to Steelwork Optimum Design", *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization - Part 3*, 1998, pp.2037~2045
10. Min-Jea Tahk, "Coevolutionary Approaches to Structural Optimization", *AIAA Journal*, Vol. 37, No. 8, 1999
11. Helio J. C. Barbosa, "A Coevolutionary Genetic Algorithm for Constrained Optimization", *Proceedings of the 1999 Congress on Evolutionary Computation*, Vol. 3, 1999
12. 조영석, 최동훈, "구속조건의 효율적인 처리를 위한 유전자 알고리즘의 개발", 대한기계학회 2000년도 춘계학술대회논문집 A, 2000, pp.725~730
13. Gen M and Cheng R, *Genetic Algorithms and Engineering Design*, Wiley-Interscience, 1996, pp.411