

혼합 유전알고리즘을 이용한 비선형 최적화문제의 효율적 해법

윤영수* · 이상용*

Abstract

This paper describes the applications of genetic algorithm to nonlinear constrained optimization problems. Genetic algorithms are combinatorial in nature, and therefore are computationally suitable for treating continuous and discrete integer design variables. For several problems, the conventional genetic algorithms are ill-defined, which comes from the application of penalty function, encoding and decoding methods, fitness scaling, and premature convergence of solution.

Thus, we develop a hybrid genetic algorithm to resolve these problems and present two examples to demonstrate the effectiveness of the methodology developed in this paper.

1. 서론

기존의 비선형최적화문제(Nonlinear optimization problem)의 해결방법들은 탐색과정의 효율성은 높였지만 제약조건이 있는 경우와 여러개의 국소최적점(Local optima)을 가지는 비선형최적화문제, 설계변수가 연속·정수·이산(Continuous, integer, discrete)형태로 혼합되어 사용되고 있는 문제, Non-convex를 지나는 목적함수의 최적화 문제에 대해서는 아직까지 그 적용상에 많은 어려움이 있다.

이에 대해 비선형 문제에 효율적으로 적용가능하며 비교적 전범위의 최적점에 근사한 해를 구할 수 있는 유전 알고리즘(Genetic Algorithms:GAs)에 의한 탐색적 방법이 최근에 활발히 연구되고 있다.(한용호, 류광렬, 1995)

* 건국대학교 공과대학 산업공학과

** 건국대학교 공과대학 산업공학과 교수

유전알고리즘은 생태계의 진화원리에 바탕을 둔 최적해에 대한 탐색 알고리즘의 하나로 다양한 종류의 문제들에 대해 효과적으로 적용가능하며 비교적 전범위의 최적점에 가까운 해를 구할 수 있다는 장점이 있다. 그러나 유전알고리즘의 적용에 있어서 제약조건이 있는 목적함수를 제약조건이 없는 목적함수로 만들기 위해 위반함수(Penalty function)를 적용하는데, 실제적으로는 어떤 위반함수를 선택하느냐에 따라 결과에 많은 영향을 미치며 또 위반계수(Penalty coefficient)를 결정하는데도 정형화된 방법이 없이 설계자의 경험에 의존하고 있다. 그리고 해의 조기수렴(Premature convergence)을 방지하기 위해 적합도 변환(Fitness scaling)을 하는데 이로 인해 계산과정은 더욱 복잡해지고 설계자가 어떤 적합도 변환방법을 선택하느냐에 따라 결과가 달라지게 되며, 최적점 부근에 수렴한 해에 대해 그 주위를 정밀하게 탐색할 방법이 없다.

본 연구에서는 이러한 문제점들과 함께 기존의 유전알고리즘이 고려하지 못한 몇몇 사항들을 개선한 혼합유전알고리즘(Hybrid Genetic Algorithms; HGA)을 설계하였으며, 사례연구에서 3개의 비선형 조합최적화문제에 HGA를 적용하여 그 유효성을 보였다.

본 연구에서 개발된 HGA는 IBM 호환기종인 Pentium-100, RAM : 16M에서 Visual Basic을 이용하여 프로그래밍(Programming)하였다.

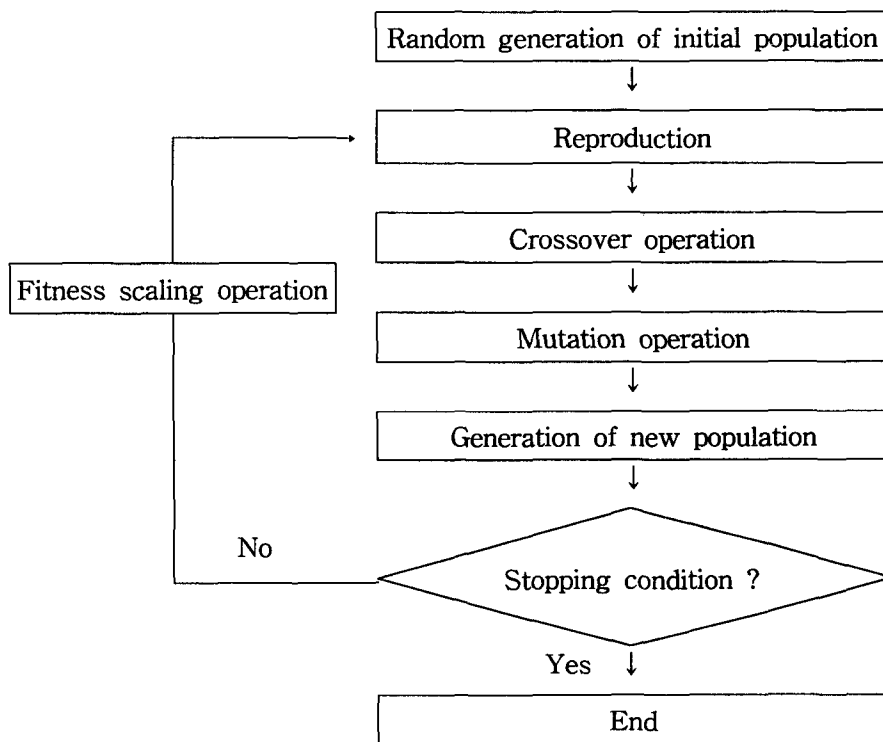
II. 본 론

1. 유전 알고리즘

유전 알고리즘은 인공지능(Artificial intelligence)의 한 기법으로서 2차원 이상의 복잡한 탐색공간에서 전범위의 최적해(Global optimal solution)를 탐색하는데 아주 효율적이며, 유연하다고 증명되어져 왔다.(Gupta, 1995) 이러한 유전 알고리즘은 생태계의 자연선택(Natural selection)과 적자생존(Survival of the fittest)에 근거를 두고 있으며, 새로운 집단(New population)을 형성할 때에 과거의 집단(Old population)에서 높은 적합도를 가지는 개체(String)가 높은 확률을 가지고 새로운 집단으로 유전한다는 것이 그 기본적인 원리이다. (Hon & Chi, 1994)

유전 알고리즘은 Holland(1975)가 그 이론적인 근거를 마련했으며, Goldberg(1989)에 의해 공학분야에서 가스 송수관문제에 대한 최적 설계가 최초로 시도된 이래 많은 발전이 되어 오고 있다.

<그림 1>은 일반적인 유전알고리즘의 흐름도를 보여 주고 있다.



<그림 1> 일반적인 유전 알고리즘의 흐름도

유전 알고리즘에 대한 기존의 연구를 살펴보면 Venkatachalan(1995)은 유전 알고리즘의 기본적인 연산자인 복제 (Reproduction), 교차변이(Crossover), 돌연변이 (Mutation)중에서 교차변이의 구조에 대해 연구·분석하였으며, Wu와 Chow(1995)는 유전알고리즘에서 사용되는 4가지 모수(Parameter)인 집단의 크기(Population size), 교차변이(Crossover), 교차변이 확률(Crossover rate), 돌연변이 확률(Mutation rate)등을 개체 자체로 처리한 Meta-GA를 개발하여 최적의 모수조합을 찾으려는 시도를 하였다. Lin과 Hajela(1992)는 이산·정수설계변수가 혼합된 설계최적화 문

제(Optimization problems with discrete and integer design variables)를 해결하는데 유전 알고리즘을 적용하였다. 또한 이 연구에서는 유전 알고리즘에서 제약조건이 있는 최적화 문제를 비제약조건의 최적화 문제로 만들기 위해 위반항(Penalty term)을 목적함수에 적용하게 되는데 이에 대한 세밀한 평가의 필요성을 강조하고 있다. Gupta(1995)는 GT(Group technology)에서 셀제조(Cellular manufacturing) 설계문제에 대해서, Hon과 Chi(1994)은 부품군(Part family)형성문제에 대해서 유전 알고리즘을 적용하였으며, Dorndore 와 Pesch(1995)는 기계일정계획문제(Machine scheduling problem)에 유전 알고리즘을 적용하였다.

2. 기존의 유전 알고리즘의 문제점 및 HGA모델 개발

기존 유전알고리즘은 단일모델(Uni-model)로 부터 다수모델(Multi-model), 조합모델(Combinatioal-model)에 이르기 까지 광범위한 영역에 걸쳐 우수해를 제공하는 것은 틀림이 없지만 유전알고리즘 실행시 위반함수법(Penalty function method), 집단 크기(Population size), 교차변이 방법(Crossover method), 교차변이율(Crossover rate), 돌연변이 방법(Mutation method), 돌연변이율(Mutation rate), 적합도 변환(Fitness scaling), 정밀도(Degree of Precision) 등에 대해서는 유전알고리즘 설계자가 임의적이고 경험적으로 적절한 방법을 사용하고 있는 실정이며, 설계자가 유전 알고리즘 실행초기단계에 이러한 방법들을 어떻게 설계하느냐에 따라 실행시간, 최적해의 수렴속도 등에 상당한 영향을 미친다.

또한 기존의 유전알고리즘은 변수설계시 0-1이진비트(Binary bit)방법을 주로 사용하는데 이 방법은 유전연산자인 교차변이와 돌연변이 수행시 상당히 간편하고 효율적이지만 각 설계변수가 이산·정수·연속변수로 혼합되어 있는 경우에는 이를 적용하기가 쉽지 않으며, 변수설계방법이 다른 대부분의 기존의 국부탐색법(Local search method)과 연계가 어렵다는 단점이 있다.

따라서 본 연구에서는 이러한 문제점을 구체적으로 지적하고 이를 개선하고자 하며 아울러 국부탐색법과의 연계성을 함께 고려한 새로운 혼합유전알고리즘(HGA)을 제시하고자 한다.

첫째, 기존에 개발된 대부분의 유전알고리즘은 설계변수가 이산·정수·연속등으로 혼합된 조합최적화문제에 대해 그 설계변수 각각을 0-1 이진비트(Binary bit)로

변경(Encoding)하여 사용한 후 적합도 평가(Fitness evaluation)에서는 다시 원래 설계변수형태로 변경(Decoding)하여 적용하고 있다. 이 경우 설계변수가 이산변수와 정수변수인 경우에는 문제가 없지만 특히 연속변수인 경우 정밀도를 적용하여 사용하는데 정밀도란 설계변수의 범위를 설계자가 임의로 정해둔 구간으로 나누는 정도를 말한다. 따라서 그 정밀도를 적게주면 빠르게 탐색을 하여 해를 구할 수 있지만 구한 해가 최적해가 아닐 경우가 있으며, 반대로 정밀도를 크게 주면 탐색할 공간이 커지게 됨으로 비록 최적해를 찾을 확률은 정밀도를 적게 준 경우보다 크지만 계산시간이 많이 걸리며 또한 단일 목적함수의 경우 최적점 부근까지는 빨리 도달하나 그 부근에서 수렴(Local convergence)하고 다목적 함수의 경우는 각 목적함수의 최적점을 정확히 구현하지 못한다는 단점이 있다.

이와같이 기존의 유전알고리즘에서 0-1 이진 비트의 사용은 유전알고리즘의 3가지 연산자인 복제(Reproduction), 교차변이(Crossover), 돌연변이(Mutation)에 대해 그 적용과 조작(Manipulation)이 간편하다는 장점이 있지만 계산과정이 복잡하고, 위에 제시된 것과 같이 서로 상이한 설계변수를 0-1이진비트로 변경하는데 어렵고 설계자의 경험에 의존하며 설계변수의 형태가 다른 최적화 탐색기법들과 조합하기가 어렵다는 단점이 있다.

이상의 기존 유전알고리즘이 가지고 있는 문제점을 개선하기 위해 본 연구에서 개발한 HGA에서는 설계변수의 형태에 대해 어떤 변형도 가하지 않고 그대로 사용하는 실수표현(Real number representation)방법을 사용한다.

둘째, 기존의 유전알고리즘은 기본적으로 제약조건이 없는 최적화 문제에 적용되며 제약조건이 있는 최적화 문제를 해결하기 위해서는 제약조건을 비제약조건으로 만든 다음 유전알고리즘을 적용하여 해결하고 있는데 이를 위해서 Goldberg(1989)는 아래의 식(1), (2)과 같은 외부위반함수인 위반상관계수(Penalty coefficient) r 을 사용하고 있으며, Lin(1992)은 아래의 식(3), (4), (5), (6), (7)에서와 같이 위반한계함수(Bounded penalty function) r 을 사용하였다. 식 (6)에서 G 는 식(1)의 위반항이 되고, $\langle g_j \rangle^2$ 은 식(2)의 Γ 과 같은 의미이다.

$$F = f - r \sum_{j=1}^p \Gamma_j \tag{1}$$

$$\Gamma_j = (\text{Max}(g_j, 0))^2 \tag{2}$$

$$F = f + \bar{F} \quad (3)$$

$$\bar{P} = G \quad \text{if } G \leq \bar{L} \quad (4)$$

$$= \bar{L} + \alpha(G - \bar{L}) \quad \text{if } G > \bar{L} \quad (5)$$

$$G = r \sum_{j=1}^b \langle g_j \rangle^2 \quad (6)$$

$$\bar{L} = pF_{av} \quad (7)$$

<기호정의>

F : 적합도 함수

f : 목적함수

r : 위반상관계수

g_j : 부등제약함수

p : g_j 의 개수

\bar{F} : 위반항(Penalty term)

F_{av} : 적합도함수의 평균

\bar{L} : 위반(Penalty)의 한계치(Limit value)

α : 선형으로 변환하는 기울기

그러나, 이러한 방법에서는 다음과 같은 문제점이 있다.

- (1) 제약 조건에 위배된 값은 위반상관계수 혹은 위반한계함수 r 에 의해서 제약조건에 접근하게 되는데 실제적인 문제에서 r 은 명목적인 충분한 정도의 값으로 각 제약조건과 목적함수의 형태에 따라 대단히 큰 값을 선택적으로 정하고 있다.

(예를들면 한용호, 류광렬(1995)의 연구에서는 r 의 값을 1,000으로 주고 있다).

- (2) α 를 정하는 것에 어떤 특정한 규칙이 없으며 문제의 유형에 따라 다르게 주고 있다.

따라서 HGA에서는 이러한 문제점을 개선하기 위해 실제 프로그래밍 과정에서 제

약조건을 모두 *if* 조건문(*if Condition*)으로 처리하여 제약조건을 만족하면 다음과정을 수행하도록 하고, 제약조건을 벗어나는 개체는 탈락시켰다.

이러한 과정은 생물체에서 일어나는 자연도태과정과 일치한다. 즉 그 환경(제약조건)에 적응하지 못하는 개체는 탈락되며, 환경에 적응하는 개체는 살아 남아서 다음 세대로 유전하게 된다.

이러한 방법을 사용할 경우, 예를들면 제약조건이 많고 복잡할때 제약조건을 벗어나는 개체가 많이 발생할 수도 있으며, 또한 전체 탐색공간중에서 해가 발견될 가능공간(*Feasible space*)이 차지하는 영역비가 적어서 탈락되는 개체가 많이 생겨 비효율적인 측면이 있을 것이라 생각되지만 실제 *if* 조건문으로 제약조건을 처리할때 프로그램 실행상 컴퓨터의 처리능력 향상으로 빠른 속도로 탐색을 실시하기 때문에 해가 발견될 가능공간이 적어도 별로 문제가 없이 다양하게 탐색을 실시할 수 있다. 따라서 제약조건이 있는 최적화 문제이더라도 상관없이 최적해를 찾으려 하였기 때문에 일정한 규칙이 없는 위반상관계수 혹은 위반한계함수 r 과 선형변환기울기 α 를 제외시켰다.

셋째, 일반적인 유전알고리즘 수행시 해의 탐색중에 몇개의 적합도가 높은 개체가 발생할 경우에 이들 개체들이 전체 집단에서 아주 큰 비율을 차지하게 되며 이것이 미성숙 수렴(*Premature convergency*)을 하게 되어 궁극적으로 구하고자 하는 최적해를 찾을 수 없는 경우가 생기며, 만일 최적해를 찾는다 하여도 그 최적해를 도출하는데 장기간에 걸쳐 수많은 세대가 흘러야 한다는 단점이 있다. 따라서 기존의 유전알고리즘에서는 구하고자 하는 최적해의 개체가 되기 위해 특별히 개체에 대한 조정이 필요하며 이를 위해 적합도 변환(*Fitness scaling*)을 하게 된다. 하지만 이러한 적합도 변환방법이 다양하고 어떠한 적합도 변환방법을 설계자가 선택·사용하느냐에 따라 최적해의 수렴정도에 상당한 차이를 보이게 된다는 제약이 있다.

따라서 HGA 에서는 이러한 문제점을 개선하기 위해 각각의 유전 연산을 거쳐 생성된 새로운 개체를 평가하여, 제약조건을 만족하지 못한다면 탈락시키고, 제약조건을 만족한다면 적합도 변환을 하지 않고 다음 단계로 바로 진행하도록 하였다.

위와 같이 적합도 변환을 하지 않을 경우에는 탐색공간에 대해 광범위한 탐색이 가능하며 따라서 해의 탐색에서 개체의 조기수렴현상을 방지하고, 개체의 다양성을 높여 수많은 세대를 거치지 않고서도 빠르게 최적해에 도달하도록 하였다.

넷째, 기존의 유전알고리즘은 확률적인 선택을 하기 때문에 다음세대의 자손

(Offspring)을 생성하는데 있어 이전집단(Old population)에 높은 적합도를 지닌 개체가 존재하더라도 다음세대에서 랜덤하게 개체가 평가되어 새로운 개체가 선택됨으로 이전집단에서 구한 최적해에 가까운 값이 다음 자손세대에 선택되지 않을 수도 있다. 즉 개체들은 프로그램상에서 임의로 발생시켜 얻은 난수를 포함하는 개체의 누적확률구간에 의해 선택됨으로 우수한 개체가 다음세대에 선택되지 못하는 경우가 생길 수도 있으며, 반면에 우수하지 않은 개체들을 다음 세대로 계속하여 유전 시킴으로서 최적해에서 멀리 떨어진 공간만을 탐색하게 되어 결국에는 최적해를 찾지 못하는 경우도 발생한다.

따라서 이러한 문제점을 개선하기 위하여 HGA에서는 이전집단에서 최고의 적합도를 갖는 하나의 개체를 의무적으로 다음 세대의 자손집단에서 선택되도록 하고 이전집단의 개체와 다음 세대의 자손집단에서 선택된 개체 전부를 정렬(Sorting)하여 높은 적합도를 가지는 순으로 세대를 선택하여 새로운 자손세대를 만드는 방법을 사용하였다.

다섯째, 기존 유전알고리즘이 복제단계에서 개체들을 임의로 난수를 발생시켜 누적확률구간에 의해 선택하는 룰렛휠(Roulette wheel)방법을 사용하는데 이 기법은 비교적 간단하지만 우수한 개체를 선택하지 못하는 경우가 생기며 이것이 유전됨으로서 최적해에서 멀리 떨어진 공간만을 탐색하게 됨으로 최종결과가 나쁘게 될 수 있다. 따라서 본 연구에서는 이러한 문제점을 개선하며, 다양한 이산·연속·정수최화문제에 효율적으로 적용하기 위해 랜덤탐색·검사알고리즘(Random search and test algorithm)을 사용한다.

랜덤탐색·검사법은 탐색공간에 대해 전범위에 걸쳐 다양하게 해를 탐색하며 비록 최적점을 찾지 못하더라도 최적점에 가까운 곳에 조기수렴하는 성질을 가지고 있기 때문에 유전알고리즘의 복제연산자에서 일반적으로 적용하고 있는 룰렛휠 방법에 의해 구한 해보다 더 나은 해를 보장할 수 있다는 장점이 있다.

여섯째, 기존의 유전알고리즘에서 사용되는 유전연산자의 교차변이에는 지금까지 많은 학자들에 의해 다양한 방법이 개발되어 사용되고 있지만 이는 설계변수가 0-1 이진비트인 경우에만 적용가능하다는 단점이 있다. 즉 설계변수의 형태가 본 연구에서 개발한 HGA의 경우처럼 실수표현형태를 가질 경우 이용이 불가능하다. 따라서 HGA에서는 Davis(1987)가 제안한 평균값 교차변이(Crossover operation by average value)를 사용한다. 이 방법은 교차변이 이전단계인 복제단계에서 해의 수

렴정도에 따라 각 설계변수들을 저장한 다음 이를 각각 평균하여 새로운 개체를 만드는 방법이다.

일곱째, 기존의 유전 알고리즘은 해의 탐색시 탐색점이 최적점 근처에 수렴할 경우에는 최적점을 포함한 탐색공간에 대한 정밀한 탐색을 실시할 방법이 없다는 단점이 있다. 하지만 HGA에서는 이러한 탐색공간에 대해 위·아래(Ups and downs)를 재탐색하는 포복법(Creeping method)을 유전연산자인 돌연변이 연산때 함께 사용하여 보다 정확한 최적점을 찾도록 하였다.

이상의 방법을 <표 1>에 간략하게 나타내었다.

본 연구에서 제시한 HGA의 수행과정은 다음을 따른다.

(1) 랜덤탐색·검사법의 적용단계

단계 1> 각 설계변수에 대해 집단의 크기(Population size)만큼 초기집단을 생성한다.

예를들어 X_1 설계변수의 탐색범위가 $2 \leq X_1 \leq 7$ 인 연속변수를 취한다면 다음과 같이 초기집단을 랜덤하게 발생시킨다.

<표 1> 일반적인 유전알고리즘과 HGA의 비교

비교방법	일반적인 유전알고리즘	HGA
변수표현형태	0-1 이진비트(binary bit)	실수표현(Real number)
복제방법	룰렛휠(Roulette wheel)	랜덤탐색·검사알고리즘(Random search and test algorithm)
비제약화 방법	위반함수법(Penalty methods)	if 조건문
적합도 변환	사용함	사용하지 않음
이전집단의 우수해 포함여부	포함하지 않음	포함
최적점부근에 대한 정밀탐색법	사용하지 않음	포복법(Creeping method)

For I = 1 to Population_Size

$$X_1(I) = (5 \times Rnd) + 2$$

Next I

여기에서 Rnd 는 랜덤발생연산자이며, $X_1(I)$ 변수는 랜덤발생에 의한 연속변수값을 취한다. 특히 이산변수의 설계일 경우 먼저 이산변수값을 정수변수값으로 대응시킨 후 발생한 정수값이 이산변수값에 대응되는 값을 취하도록 HGA를 설계하였다.

단계 2> 단계 1>에서 발생한 집단을 제약조건에 대입시킨 후 제약조건을 만족하지 못한 개체군들은 탈락시키고, 제약조건을 만족하는 개체군들은 전부 목적함수에 대입하여 해를 구한다.

단계 3> 단계 1,2>를 반복한 후 전범위 최적해에 가까운 국소최적점을 찾는다.

(2) 평균값에 의한 교차변이 적용단계

단계 4> 단계 1,2,3>에 의해 국소최적점에 수렴되어지는 과정에서 발생하는 각 설계변수들의 개체군들 상호간의 평균값에 의한 교차변이 집단을 생성한다.

평균값에 의한 교차변이 단계는 각각의 두 부모개체를 평균하여 한개의 새로운 자손(Offspring) 개체를 만든다. 만일 단계 1,2,3>에서 전체 10번의 수렴과정을 거칠 경우 각 설계변수에 대해 9개의 자손이 만들어지게 된다.

예를들어 <그림 2>와 같은 집단이 단계 1, 2, 3>에서 생성되었다면 평균값에 의한 교차변이의 결과는 <그림 3>과 같으며, <그림 2>의 첫번째의 X_1 개체와 두 번째 발생한 X_1 개체를 평균한 12.2225를 <그림 3>의 X_1 자손개체로 생성한다.

$X_1 = 12.2325,$	$X_2 = 5.789,$	$F(X) = 270.237$
$X_1 = 12.2125,$	$X_2 = 5.353,$	$F(X) = 269.211$
$X_1 = 12.2015,$	$X_2 = 5.301,$	$F(X) = 268.200$
.	.	.
.	.	.
.	.	.
.	.	.

<그림 2> 랜덤 · 탐색검사법의 수렴과정

$X_1 = 12.2225,$	$X_2 = 5.710,$	$F(X) = 268.212$
$X_1 = 12.2070,$	$X_2 = 5.327,$	$F(X) = 265.312$
.	.	.
.	.	.
.	.	.

<그림 3> 평균값 교차변이 결과

이러한 과정을 각각의 설계변수에 대해 전부조합하여 수행한다.

단계 5> 단계 4>에서 발생한 새로운 자손집단을 제약조건에 대입한후 제약조건을 만족하면 목적함수에 대입하여 단계 3>에서 구한 최적점보다 더 우수한 해가 나오면 이를 저장한다. 만족하지 못하면 집단을 탈락시킨다.

(3) 돌연변이 및 포복법(Creeping method)의 적용단계

단계 6> 단계 5>에서 구한 해에 대해 돌연변이 연산과 포복법을 실시한다.

돌연변이 연산은 기존의 집단에 대해 돌연변이 확률을 적용하여 임의의 탐색공간에서 발견될지도 모르는 최적값을 탐색하는 절차이다. 또한 기존의 연구에서 적용하지 않은 포복법을 돌연변이 연산과 함께 사용한다.

포복법은 최적점 부근에 수렴한 국소최적점을 개선하기 위한 방법으로 최적해가 포함된 탐색공간에 대해 국소최적점 주위를 위·아래로 정밀하게 탐색하여 전범위 최적점을 찾기 위한 방법이다.

HGA에서는 설계자가 정해진 포복범위의 값을 이용하여 포복법을 실시한다.

포복법에 의한 탐색의 경우 예를들어 X_1 의 값이 11.5이고 설계자가 정해진 포복값(Creeping_value)이 1.5라면 X_1 은 10.0 ~ 13.0 범위에 대해 재탐색을 실시한다. 이에 대한 변수설계는 다음과 같다.

```

For I = 1 to Population_Size
    Temp = (2 × Creeping_Value × Rnd) - Creeping_Value
     $X_1(I) = C\_X1\_Value + Temp$ 
Next I
    
```

여기에서 Temp는 임시기억장소이며, C_X1_Value는 교차변이에서 구해진 국소최

적 X_1 의 값이다.

단계 7> 단계 6>에서 구해진 각 설계변수값을 제약조건에 대입하여 만족하면, 이를 목적함수에 대입하여 교차변이에서 구한 최적점보다 더 나은 해가 나오면 이를 저장한다. 만일 제약조건을 만족하지 못하면 탈락시킨다.

단계 8> 설계자가 정해진 중단조건(Termination Criterion)에 도달할 때 까지 단계 6,7>을 반복한다.

이상과 같은 절차를 통해 기존의 유전알고리즘모델 적용시 문제점으로 지적되었던 0-1이진비트 변환문제, 적합도 변환문제, 정밀도 정도문제, 위반함수문제, 조기수렴에 의한 국소최적점 문제 등을 개선한 HGA모델을 설계했다.

이에 대한 흐름도(Flowchart)는 <그림 4>와 같다.

3. 사례연구

본 연구에서 제시하고자 하는 모델은 총 3개로서 첫번째 모델(Model 1)은 Gen(1994)등이 제시한 Sin함수를 적용한 모델로서 비교적 간단한 비선형문제이다. 두번째 제시한 모델(Model 2)은 Sandgran(1990)이 제시했으며, 압력용기(Pressure vessel)의 최적설계를 위하여 설계변수들이 연속, 이산값으로 혼합된 조합최적화모델이다. 마지막으로 제시할 모델(Model 3)은 강화 콘크리트 빔(Reinforced concrete beam) 최적설계문제로 Aamir and Hasegawa(1989)가 제시했으며 설계변수가 연속·정수·이산값으로 혼합된 조합최적화 문제이다.

이와 같이 제약조건이 있는 비선형최적화 모델들을 이용해 기존에 개발된 최적화 기법들과 유전 알고리즘을 본 연구에서 개발된 HGA와 비교·분석하였다.

(1) Model 1 (Sin 함수)

Objective function

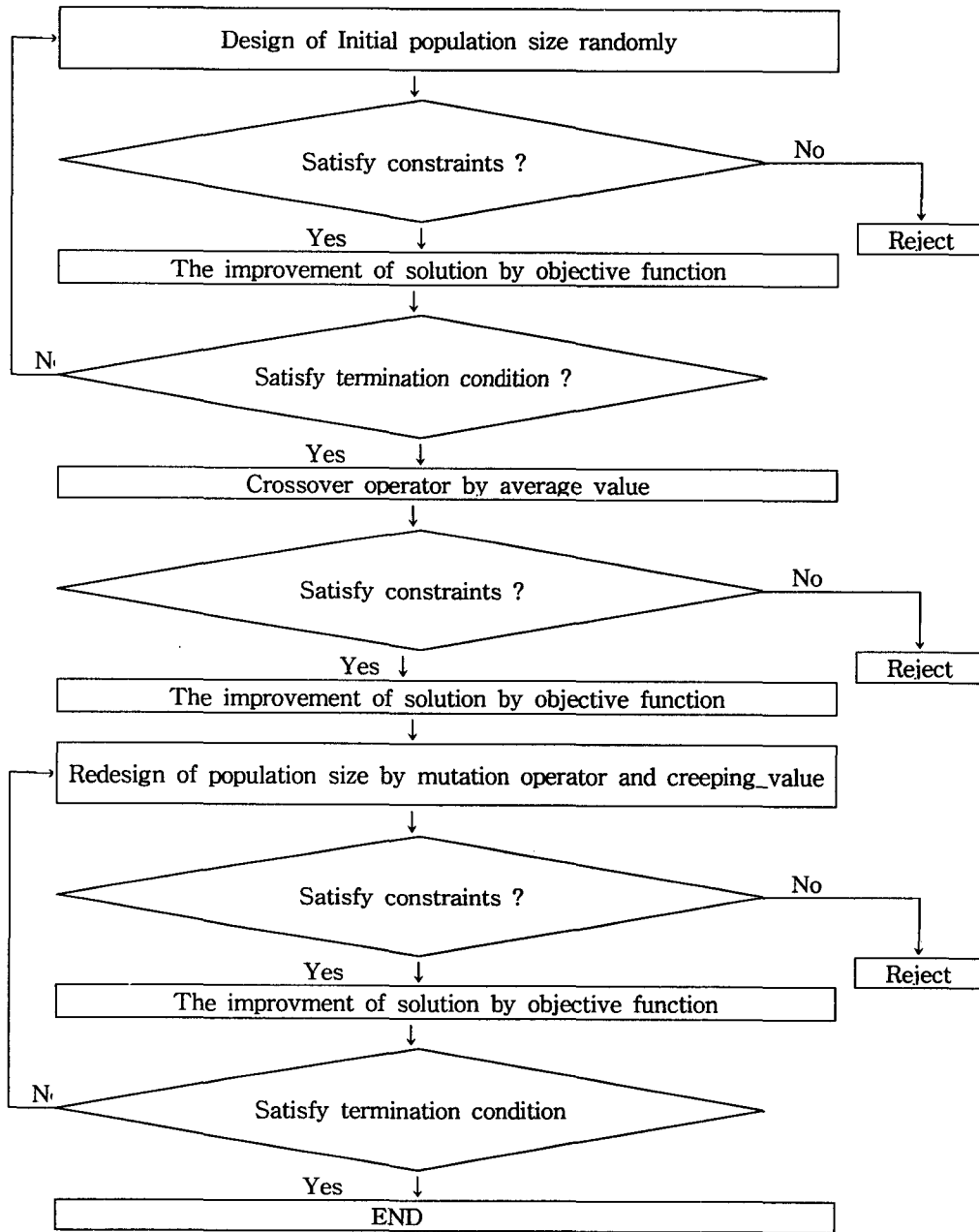
$$\text{Maximize } f(x) = 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2)$$

Subject to

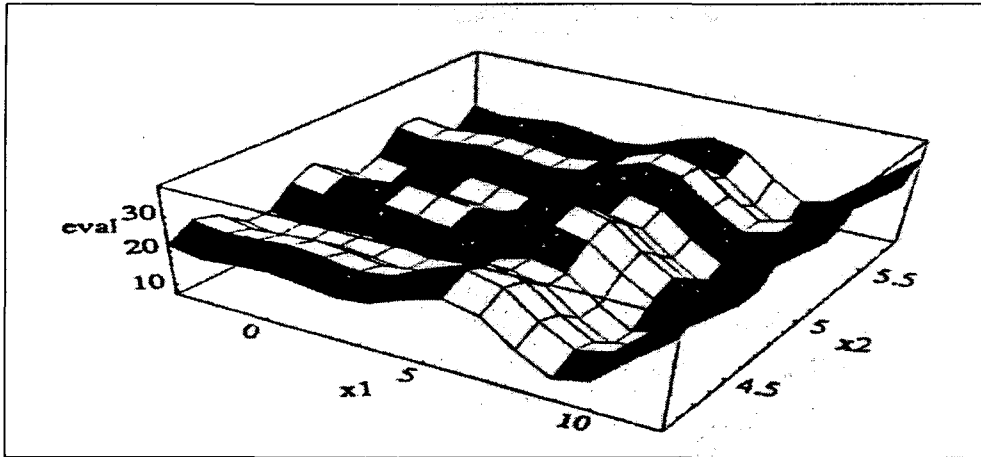
$$-3 \leq x_1 \leq 12.1$$

$$4.1 \leq x_2 \leq 5.8$$

위의 문제의 실제 탐색공간은 <그림 5>와 같으며 이를 기존의 유전 알고리즘 연구와 HGA로 실행한 결과의 비교를 <표 2>에 나타내었다.



<그림 4> HGA의 수행 흐름도



<그림 5> Model 1의 탐색공간

<표 2> Model 1의 실행결과비교

	Davis(1981)	Gen et al(1994)	김창은(1994)	This paper	Type of Variables
Methods	GA	GA	GA	HGA	
Population size	20	3	10	50	
Crossover rate	0.25	0.25	0.25		
Mutation rate	0.01	0.01	0.01	0.15	
Generation No.	1000	667	419	37	
Creeping Value				0.1	
x_1	9.623693	11.6141	11.631407	11.632000	Contonuous
x_2	4.427881	5.7300	5.724824	5.728000	"
$f(x)$	35.477938	38.4539	38.818208	38.858776	

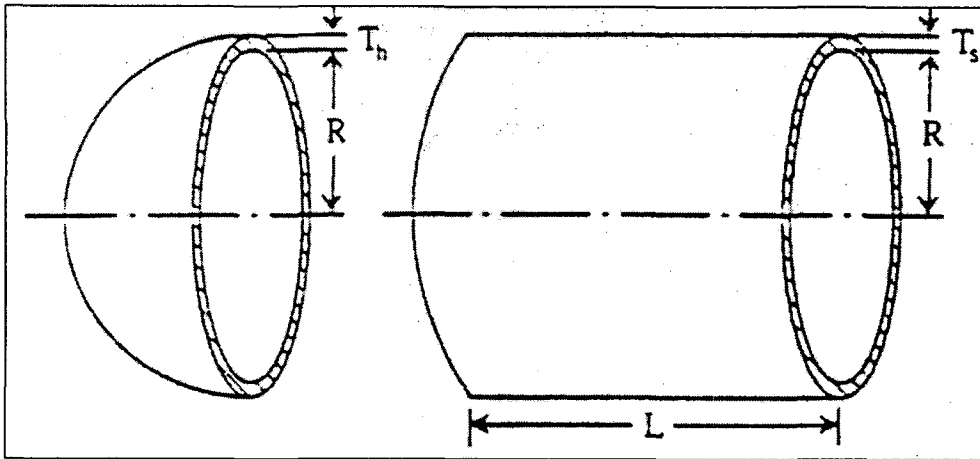
<표 2>를 살펴보면 Davis, Gen 등과 김창은 등의 연구결과는 기존의 유전 알고리즘을 이용하여 해를 구하고 있지만 두 연구 모두 더 이상 해를 개선시키지 못하고 국소 최적점에 빠졌다. 그러나 HGA는 이러한 현상을 개선하였기 때문에 더 우수한 해를 도출하고 있음을 볼 수 있으며 집단의 크기는 기존의 연구들보다 증가하였지만 세대수(Generation No.)에서는 Davis보다는 27배, Gen등의 연구보다 18배,

김창은 등의 연구보다는 11배정도로 더 빠르고, 우수한 결과를 도출하고 있다.

(2) Model 2 (*Design of a Pressure Vessel*)

압력용기(Pressure Vessel) 구조는 Sandgren(1990)이 제시했으며 <그림 6>에 나타나 있다. 설계변수는 용기의 사양에 필요한 치수이며, 다음과 같이 표시된다.

$$\text{Find } X = [T_s, T_h, R, L]^T = [x_1, x_2, x_3, x_4]^T$$



<그림 6> 압력용기의 구조

목표함수는 압력용기의 총제조비용을 최소화하는 것이다. 설계변수 x_1, x_2 는 이산변수(Discrete variables)이고, 0.0625 정수배수(Integer multiples)를 가지며, x_3, x_4 는 연속변수(Continuous variables)이다.

이에 대한 수리적 모델은 아래와 같다.

Objective function

Minimize

$$F(x) = 0.6224 x_1 x_3 x_4 + 1.7781 x_2 x_3^2 + 3.1661 x_1^2 x_4 + 19.84 x_1^2 x_3$$

Subject to

$$G_1(X) = x_1 - 0.0193x_3 \geq 0$$

$$G_2(X) = x_2 - 0.00954x_3 \geq 0$$

$$G_3(X) = \pi x_3^2 x_4 + \frac{4}{3} \pi x_3^3 - 750 \times 1728 \geq 0$$

$$G_4(X) = -x_4 + 240 \geq 0$$

$$G_5(X) = x_1 - 1.1 \geq 0$$

$$G_6(X) = x_2 - 0.6 \geq 0$$

이상과 같은 이산·연속변수가 서로 혼합된 조합최적화 문제에 HGA를 적용하여 기존의 연구들과 비교·분석한 결과는 <표 3>과 같다.

<표 3> Model 2의 실행결과·비교

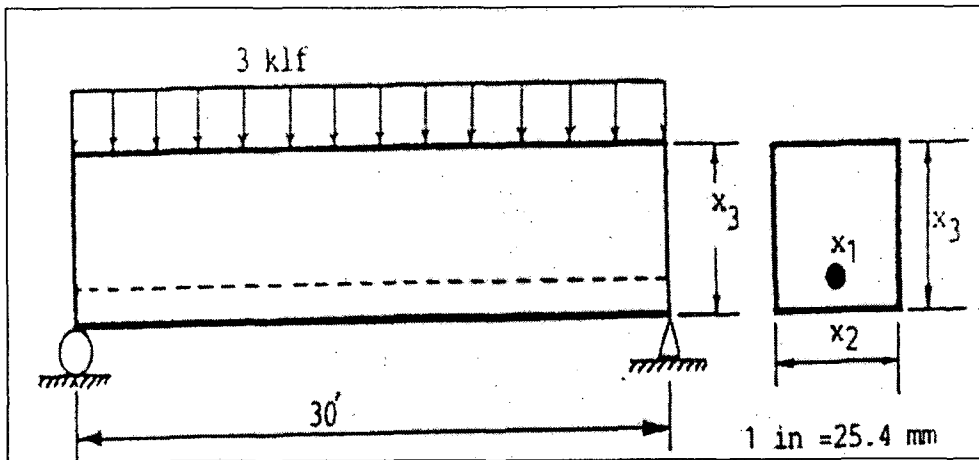
	Fu et al. (1991)	Sandgren (1990)	This paper	Type of Variables
Methods	IDCNLP	Branch and Bound	HGA	
x_1	1.125	1.125	1.125	discrete
x_2	0.625	0.625	0.625	discrete
x_3	48.3807	48.97	58.2901	continuous
x_4	111.7449	106.72	43.6930	continuous
F(X)	8048.619	7982.5	7197.73	

<표 3>의 Fu et al.(1991)의 결과는 IDCNLP(Integer-discrete-continuous non-linear programming) 알고리즘을 이용하여 해결하였는데, 이 방법은 일반적인 비선형계획법의 해결절차에 근거를 두고 있으며 문제 해결과정에서 초기입력변수의 값을 선택하는데 어려움이 있다. Sandgren(1990)의 결과는 분지한계법(Branch and bound)을 이용하여 해를 구하고 있지만 적용기법의 특성상 초기상태에서는 전체 마디(node)의 수를 알 수 없기 때문에 프로그램 작성시에 실제 필요한 저장공간보다 더 많은 저장공간을 미리 확보하여야 하며, 문제의 규모가 커질 경우 해법상 해결과정이 복잡하기 때문에 국소수렴이 될 가능성이 크다.

반면에 본 연구의 HGA(집단의 크기 : 30, 돌연변이율 : 0.15, 세대수 : 5000, 포복값: 0.1)에서는 세대수가 1721번째에서 $F(x) : 7197.73$ 으로 기존의 F_u 등과 Sandgren이 구한 해 보다 더 우수한 해를 구하였다. 이러한 현상은 기존의 두 연구 모두 국부해로 수렴하였지만 HGA는 해의 다양성을 실현하고 미성숙 수렴현상을 방지하였기 때문에 비교적 빠른 시간내에 수많은 세대를 거치지 않고서도 더 우수한 해를 구할 수가 있었다.

(3) Model 3 (*Design of Reinforced Concrete Beam*)

강화 콘크리트 빔(Reinforced concrete beam)구조는 Armir and Hasegawa(1989)가 제시했으며 <그림 7>에 나타나 있다.



<그림 7> 강화 콘크리트 빔의 구조

X_1 설계변수는 강화바(Reinforcing Bar)의 교차부분영역(Cross sectional Area)이며, <표 4>에 정의된 이산값을 가진다. 콘크리트 빔의 깊이를 결정하는 X_2 는 정수값을 가진다. 설계변수 X_3 는 빔의 깊이를 정의하고 있으며, 연속변수의 값을 가진다.

<표 4> 강화 콘크리트 빔의 이산설계공간

X ₁	Bar type	Area(in ²)	X ₁	Bar type	Area(in ²)	X ₁	Bar type	Area(in ²)
1	1#4	0.20	27	8#5	2.48	53	9#7	5.40
2	1#5	0.31	28	13#4	2.60	54	7#8	5.53
3	2#4	0.40	29	6#6	2.64	55	13#8	5.72
4	1#6	0.44	30	9#5	2.79	56	10#7, 6#9	6.00
5	3#4, 1#7	0.60	31	14#4	2.80	57	14#6	6.16
6	2#5	0.62	32	15#4, 5#7, 3#9	3.00	58	8#8	6.32
7	1#8	0.79	33	7#6	3.08	59	15#6, 11#7	6.60
8	4#4	0.80	34	10#5	3.10	60	7#9	6.90
9	2#6	0.88	35	4#8	3.16	61	9#8	7.11
10	3#5	0.93	36	11#5	3.41	62	12#7	7.20
11	5#4, 1#9	1.00	37	8#6	3.52	63	13#7	7.80
12	6#4, 2#7	1.20	38	6#7	3.60	64	10#8	7.90
13	4#5	1.24	39	12#5	3.72	65	8#9	8.00
14	3#6	1.32	40	5#8	3.95	66	14#7	8.40
15	7#4	1.40	41	9#6	3.96	67	11#8	8.69
16	5#5	1.55	42	4#9	4.00	68	15#7	9.00
17	2#8	1.58	43	13#5	4.03	69	12#8	9.48
18	8#4	1.60	44	7#7	4.20	70	13#8	10.27
19	4#6	1.76	45	14#5	4.34	71	11#9	11.00
20	9#4, 3#7	1.80	46	10#6	4.40	72	14#8	11.06
21	6#5	1.86	47	15#5	4.65	73	15#8	11.85
22	10#4, 2#9	2.00	48	6#8	4.74	74	12#9	12.00
23	7#5	2.17	49	8#7	4.80	75	13#9	13.00
24	11#4, 5#6	2.20	50	11#6	4.84	76	14#9	14.00
25	3#8	2.37	51	5#9	5.00	77	15#9	15.00
26	12#4, 4#7	2.40	52	12#6	5.28			

이에대한 수리 모델은 다음과 같다.

Minimize

$$F(X) = 29.4X_1 + 0.6X_2X_3$$

Subject to

$$G_1(X) = X_1X_3 - 7.735\left(\frac{X_1^2}{X_2}\right) - 180 \geq 0$$

$$G_2(X) = 4.0 - \left(\frac{X_3}{X_2}\right) \geq 0$$

Where

- X_1 : Discrete
- X_2 : + continuous
- X_3 : + integer

이상과 같은 이산·정수·연속변수가 서로 혼합된 조합최적화 문제에 HGA를 적용하여 기존의 연구들과 비교·분석한 결과를 <표 5>에 제시하였다.

<표 5> Model 3의 실행결과·비교

	Amir & Hasegawa (1989)	This paper	Type of Variables
Methods	IGD	HGA	
x_1	7.8	6.16	Discrete
x_2	31	35	Continuous
x_3	7.79	8.750089	Integer
F(X)	374.20	364.85588	

<표 5>의 결과를 보면 Amir & Hasegawa는 정수 경사방향법(Integer gradient direction; IGD) 기법을 사용하여 문제를 해결하였다. 이 기법은 원래 이산변수를 처리하기 위해 설계된 Rosenbrocks 직각절차(Rosenbrocks orthogonalization procedure; ROP)의 확장 형태이며, 문제 해결과정에서 연속변수를 아주 작은 값의 증가치를 가지는 이산변수로 취급한다는 개념을 이용하고 있다. 따라서 연속변수의 특성을 유지하면서도 이산변수와 같이 처리할 수 있다. 하지만 이 기법은 설계문제에서 특히 정밀성을 요구하는 연속변수를 이산변수화시켜 처리하고 있기 때문에 정확한 의미의 연속변수값을 구할 수 가 없으며, 따라서 이 기법으로 구한 결과는 최적해에 근사한 적정해일 수 밖에 없다. 하지만 본 연구에서 제시한 HGA는 설계변수들에 대해 어떠한 변화를 주지 않고 그 자체를 그대로 처리하여 문제를 해결하고 있기 때문에 좀더 우수한 해를 구할 수 있다.

III. 결 론

일반적으로 유전 알고리즘은 설계변수의 형태가 이산·연속·정수등으로 혼합되어 있는 비선형최적화문제에 있어서 기존에 개발된 최적화 기법보다 그 적용범위와 해결능력이 뛰어나다고 증명되어 왔다. 하지만 기존의 유전 알고리즘은 집단의 수, 반복수, 복제방법, 위반함수법, 적합도 변환방법, 교차변이법, 교차변이 확률, 돌연변이법, 돌연변이 확률 등에 대해 설계자가 경험적으로 지정해 주어 문제를 해결하고 있다. 즉 그 결정에 대해 정형화된 규칙이 없고, 문제에 따라 달리 적용하여야 한다. 따라서 본 연구에서는 이러한 문제점을 개선한 혼합 유전알고리즘(HGA)을 설계하였다.

먼저 설계변수표현방법에 있어서 그 절차가 복잡한 기존의 0-1 이진비트(binary bit) 보다 더 간편한 실수표현방법을 사용하였고, 제약조건 자체를 if 조건문으로 처리하여 위반함수의 도입을 배제하였으며, 각 연산마다 개체를 평가하여 적합도 변환 없이 집단의 다양성을 시도하였다. 또한 복제단계에서 자주 사용되는 룰렛휠 방법보다 더 효율적인 랜덤탐색·검사알고리즘을 대체하여 사용하였으며, 최적점 부근에 대한 정밀한 탐색을 위해 기존의 유전알고리즘에서 사용하지 않는 포복법을 돌연변이 연산과 함께 사용하였다.

사례연구에서는 3가지의 비선형최적화 문제에 기존의 최적화 기법 및 유전알고리즘과의 비교를 통해 본 연구에서 개발한 HGA을 유효성을 입증하였다.

하지만 HGA 역시 설계자가 경험적으로 정해 주는 반복수, 집단의 크기, 돌연변이율, 포복값 등의 범위를 어느 정도로 하느냐에 따라 개체의 다양성 및 그 최적해의 수렴속도 등에 영향을 주게된다.

이러한 이유로 유전 알고리즘에 관계된 모든 값들을 서로 조합적으로 연관을 시켜 실행할 수 있는 종합적인 체계를 구축하는 것이 필요하다고 하겠다.

참 고 문 헌

- 김기화, "Genetic Algorithm에 의한 다목적함수 최적구조설계," 서울대학교 조선해양공학과 박사학위 논문, (1993)
- 김창은, 주용준, "Application of Genetic Algorithm for Simple Parameter Optimization Problem," 추계 산업공학회 학술발표대회, 9, PP. 27-34, (1994)
- 류재구, "유전자 알고리즘의 개념과 프로그래밍", *Computer Magazine*, . 10, (1993)
- 이동곤, 김수영, 이창익, "최적화의 효율향상을 위한 유전해법과 직접탐색법의 혼용에 관한 연구," 산업공학, Vol. 8, No. 1, 3, (1995)
- 한용호, 류광렬, "기계-부품군 형성문제의 사례를 통한 유전 알고리즘의 최적화 문제에의 응용," 경영과학, Vol. 12, No. 2, 8, (1995)
- 玄 光男, 田口雄章, 怒木望, "Mathematica による 遺傳的 アルゴリズム: 非線形最適化その1 : geneOpt-1/math," *ハソコリテラシ*, 第19卷第5號, PP. 34-49, (1994. 7)
- Amir, H., Hasegawa, T., "Nonlinear Mixed-Discrete Structural Optimization," *Journal of Structural Engineering*, Vol. 115, No. 3, PP. 626 - 646, (1989)
- Davis. L., *Handbook of Genetic Algorithms*, New York : Van Nostrand Reinhold, (1981)
- Davis, L. and Steenstrup, M., *Genetic algorithms and simulated annealing : an overview*, in L. Davis (ed.) (Morgan Kaufmann, Los Altos), (1987)
- Dorndore U., and Pesch E., "Evolution Based Learning in a Jopshop Scheduling Environment," *Computers Ops. Res.*, Vol. 22, No. 1, PP. 25-40, (1995)

Fang, H. L., Ross, p. and Corne, D., "A Promising Genetic Algorithm Approach to Job-shop Scheduling, Re-Scheduling, and Open-Shop Scheduling Problems," *Proceeding of ICGA Conference*, PP. 375-382, (1993)

Fu J. F., Fenton, R. G., and Cleghorn, W. L., "A Mixed Integer - discrete - Continuous Programming Method and its Application to Engineering Design Optimization," *Engineering Optimization*, Vol. 17, PP. 263-280, (1991)

Goldberg, D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Reading, Massachusetts, Addison-Wesley, (1989)

Grefenstette, J. J., "Optimization of control parameter for genetic algorithm," *IEEE Transactions of systems, man and cybernetics*, Vol. 16, No. 1, (1986)

Gupta, Y. P., Gupta, M. C., Kumar, A. K. and Sundram, C., "Minimizing total intercell and intracell moves in cellular manufacturing: a genetic algorithm approach," *INT. J. of Computer Integrated Manufacturing*, Vol. 8, No. 2, PP. 92-101, (1995)

Holland, J. H., *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan, The University of Michigan Press, (1975)

Hon, K. K. B. and Chi, H., "A New Approach of Group Technology Part Families Optimization," *Annals of the CIRP*, Vol. 43, 1, (1994)

Lin, C. Y. and Hajela P., "Genetic Algorithms in Optimezation Problems with Discrete and Integer Design Variables", *Eng. Opt.*, Vol. 19, PP. 309-327, (1992)

Sandgran, E., "Nonlinear integer and discrete programming inmechanical design optimization," *ASME J. Mech. Des.*, Vol. 112, No. 2, PP. 223-229, (1990)

Venkatachalam A. R., "An Analysis of an Embedded Crossover Scheme on a GA-Hard Problem," *Computers Ops. Res.*, Vol. 22, No. 1, PP. 149-157, (1995)

Wu, S. J. and Chow, P. T., "Genetic Algorithms for Nonlinear Mixed Discrete-Integer Optimization Problems via Meta-Genetic Parameter Optimization," *Eng. Opt.*, Vol. 24, PP. 137-159, (1995)