

비선형 최적화문제의 해결을 위한 개선된 유전알고리즘의 연구

우병훈* · 하정진**

A Study on Improved Genetic Algorithm to solve
Nonlinear Optimization Problems

Bung-hun Woo* · Jung-jin Ha**

ABSTRACT

Genetic Algorithms have been successfully applied to various problems (for example, engineering design problems with a mix of continuous, integer and discrete design variables) that could not have been readily solved with traditional computational techniques.

But, several problems for which conventional Genetic Algorithms are ill defined are premature convergence of solution and application of exterior penalty function.

Therefore, we developed an Improved Genetic Algorithms(IGAs) to solve above two problems. As a case study, IGAs is applied to several nonlinear optimization problems and it is proved that this algorithm is very useful and efficient in comparison with traditional methods and conventional Genetic Algorithm.

1. 서 론

최적화 기법은 기존의 많은 연구들을 통해서 개발되어 왔고 그 해법 또한 다양하게 적용되고 있다. 즉 여러가지 제약조건들을 모두 만족해야 하는 설계변수들의 집합중에서 최적해를 구하는 방법에 대해서는 기존의 연구들을 통해서 정형화된 기법들이 개발되어져 왔다. 그러나 이러한 최

적화 기법중에서 설계변수들이 정수(Integer), 실수(Real number) 또는 이산치(Discrete value)를 함께 가지며 제약조건과 목적함수에 서로 혼합되어 최적해를 구하는 조합 최적화 문제(Combinational optimal problem), 여러개의 국소최적점(Local optimum)이 존재하는 경우에 전범위 최적점(Global optimum)을 구하는 문제, 그리고 다목적함수(Multi-objective function)의

* 대구공업전문대학 공업경영과

** 동아대학교 산업공학과

설계시 최적해 집합의 효율적인 구현문제들에서는 아직까지 효과적으로 적용할 수 있는 해법이 개발되지 않고 있는 실정이다.

이러한 문제점을 비교적 쉽게 해결하여 줄 수 있는 인공지능(Artificial intelligence)의 한 기법인 유전알고리즘(Genetic algorithm)이 최근에 활발히 연구되고 있다. [4]

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. 하지만 기존의 유전알고리즘은 설계시 고려되고 있는 여러가지 변수들, 즉 설계변수의 출발점, 탐색의 폭, 실행 종료조건, 외부위반함수법(Exterior penalty function method)의 도입에 대한 기법이 아직까지 정형화되지 않은 상태이며, 해의 탐색 도중에 최적해에 도달하지 못하고 국소최적점(Local optimum)에 미성숙 수렴(Premature convergence)되어 전범위 최적해(Global optimal solution)를 구하지 못하는 경우도 있다. 따라서 본 연구에서는 이러한 기존의 유전알고리즘 구축시 문제가 되고 있는 외부위반함수법의 도입문제와 미성숙 수렴현상을 개선한 유전알고리즘(Improved genetic algorithm ; IGA)을 제시하였으며, 이를 3가지의 비선형 최적화문제에 적용하여 본 연구에서 개발한 IGA의 유효성을 입증하고자 한다.

본 연구에서 개발된 IGA는 IBM 호환기종인 486 DX2-50, RAM : 4M 에서 C Language (Turbo-C, Version 2.0)를 이용하여 프로그래밍(Programming)하였다.

2. 유전알고리즘 (Genetic Algorithms)

유전 알고리즘은 생태계의 자연선택(Natural selection)과 적자생존(Survival of the fitness)의 원리를 이용한 최적화 기법의 한 방법으로 새로운 집단(New population)을 형성할 때에 이전 집단(Old population)에서 높은 목적함수값을 가지는 개체(String)가 하나 또는 그 이상의 후손(Offspring)을 만드는데에 더 높은 확률을 가지고 개체변이를 한다는 것이 기본적인 원리이다. [10] 이러한 유전알고리즘은 John Holland[10]와 그의 연구진들이 유전알고리즘에 대한 연구를 종합, 체계화 시켰으며, 공학분야에서는 Goldberg[7]에 의해 가스 송수관문제에 대한 최적 설계가 최초로 시도된 이래 많은 발전이 되어 오고 있다.

유전알고리즘은 2차원 이상의 복잡한 탐색공간에서 전범위의 최적해를 효율적으로 탐색하는 기법이기 때문에 많은 분야에서 아주 효과적이고 유연하다고 증명되어져 왔으며[9], 다음과 같은 특징이 있다. [9][16]

(1) 모수의 코드(Code)를 사용

유전알고리즘은 모수 자체를 사용하는 것이 아니라 유한의 길이를 가지는 개체로 표현되는 2진수(Binary digit number) 코딩을 이용한다. 즉, 이산적인 표현이 되는데 이러한 특성은 계산하는데에 있어서 이산변수, 정수변수를 다룰 수 있게 하고, 연속변수인 경우에는 특정한 정밀도(Precision)를 이용하여 계산할 수가 있기 때문에 변수와 목적함수의 형태에 제약받지 않는다.

(2) 다수 탐색점의 동시탐색

기존의 많은 최적화 기법에서는 탐색점을 결정하기 위해 개개점(Point-to-point)방법을 사용하는데, 이것은 여러개의 가능해(Multi feasible solution)중에서 최적해를 찾지 못하고 국부적인(Local) 해에 국한되는 위험성을 가지고 있다. 하지만 유전알고리즘에서는 한 점의 탐색점이 아닌 여러 탐색점의 집단(Population)에서 출발하여 적절한 다양성(Diversity)을 유지하면서 우수한 후보해들간의 정보를 교환하기 때문에 전범위의 최적해에 도달할 확률이 매우 크다.

(3) 부가적인 정보의 불필요

기존의 최적화 방법들에서는 도함수를 유도해야 하는 것과 같이 문제에 대해 부가적인 정보를 필요로 한다. 그러나, 매번 문제가 주어질 때마다 필요한 지식을 이끌어 낸다는 것이 항상 용이하지만은 않으며 경우에 따라서는 불가능하기도 하다. 하지만 유전알고리즘은 각 후보해에 대해 제약조건을 고려한 목적함수의 값만을 필요로 하고 일체의 다른 정보를 요구하지 않으므로 매우 광범위한 적용이 가능하다.

(4) 다른 기법과의 혼용

유전알고리즘의 전체탐색 능력과 동시탐색 능력을 좀 더 강력한 수렴특성을 갖는 다른 지역적 최적화 기법과 조합하여 혼합 알고리즘(Hybrid genetic algorithm)을 구축할 수가 있다.

(5) 다양한 변수설계

유전알고리즘은 고유영역(Domain-independent)

에서 실행되기 때문에 특정한 변수설계 문제에서 아주 적은 정보만 있으면 된다. 이것은 다양한 최적화문제를 취급할 수 있게 해 준다.

(6) 효과적인 탐색기법 (Robustness)

미적분학(Calculus-based method) 방법을 이용한 탐색에서는 아주 어렵고 복잡하지만 유전알고리즘은 비연속적인 변수설계 공간에서도 강력한 탐색 능력이 있다.

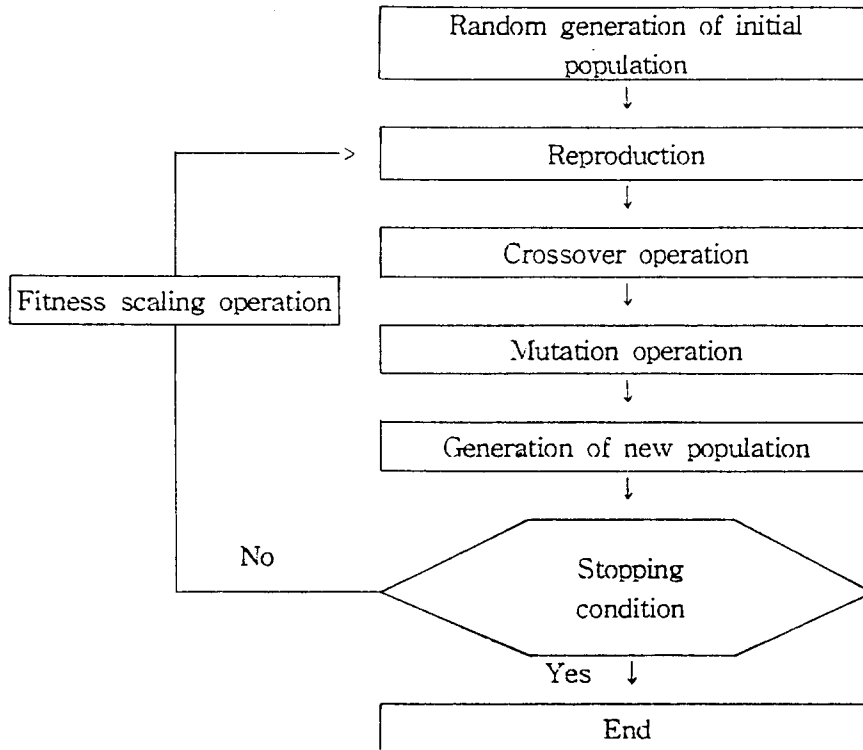
(7) 확률적 변이 규칙 (Probabilistic transition rule)

유전알고리즘은 확정적인 규칙(Deterministic rule)을 사용하는 것이 아니라 무작위 선택(Random selection)을 도구로 하는 확률적인 변이 규칙을 사용하고 있다.

이러한 유전알고리즘의 구현 절차는 먼저 제약조건을 만족하는 설계변수의 가능해 초기집단(Initial population)을 무작위로 발생시키고 이들 각각을 목적함수에 대입하여 1세대의 최적해를 구한다. 여기서 구해진 최적해를 유전 연산자(Genetic operator)인 복제(Reproduction), 교차변이(Crossover), 돌연변이(Mutation)등을 통해 적합도 함수(Fitness function)에 의해 평가되고 다음 세대의 집단인 자손(Offspring)을 형성한다. 형성된 자손의 집단을 다시 목적함수에 대입하여 2세대의 최적해를 구한다. 이러한 과정을 계속 반복하여 최적해가 나오면 중지한다. 그리고 유전알고리즘의 반복되는 절차를 종료하는 조건(Stopping condition)에는 반복되는 세대수를 프로그램설계상에서 미리 설정해 주는 방법과 목적함수 값이 인정할 만한 어느 정도의 수준에 도달하면

종료하는 방법등이 일반적으로 많이 사용된다.
 <그림 1> 은 유전알고리즘의 일반적인 계산절

차를 나타내고 있으며, 유전알고리즘에 관한 최근
 의 연구들을 살펴보면 다음과 같다.



<그림 1> 일반적인 유전 알고리즘의 흐름도

Wu와 Chow[16]는 이산변수와 정수변수가 혼합된 최적화문제에서 기존의 유전알고리즘을 개선시킨 Meta genetic algorithm을 제시하였으며, Rajeev[14]는 이산화변수(Discrete sizing variable)를 이용하여 구조적 최적화 문제(Structural optimization problem)에 대한 유전알고리즘의 적용을 연구하였고, Lin[12]은 이산치와 정수가 혼합된 최적설계문제(Mixed discrete-integer optimal design problem)에 적합한 유전알고리즘을 개발하였다. Grefenstette[8]는 유전알고리즘에 의한 최적설계시에 사용되는 설계상수의 여러 가지 조합에 대해 몇몇 수학적 함수를 대상으로

최적화를 행하여 그 활용도에 관한 연구를 하였고, Gupta[9]는 GT(Group technology)에서의 셀제조(Cellular manufacturing) 설계문제에 대해 적용하였으며, Hon과 Chi[11]는 GT에서의 부품군(Part family) 형성문제에 대해서 연구하였고, Fang등[5]은 Job shop scheduling의 최적화문제에 대하여 유전알고리즘의 적용사례를 발표하였다.

이동곤 등[3]은 기존의 유전알고리즘이 해의 도출과정에서 조기수렴 현상이 나타나는바 이를 수정하여 직접탐색법(Direct search method)과 유전알고리즘을 결합한 혼합형(Hybrid) 최적화

방법을 제시하였다. 김기화[1]는 비선형 단일목적함수와 다목적함수에 유전알고리즘을 적용하였으며, 한용호 등[4]은 기계-부품균형성문제에 유전알고리즘을 적용하였다.

3. 기존의 유전 알고리즘의 문제점과 개선된 유전알고리즘(IGA)의 설계

본 연구는 다음의 3가지 측면에서 기존의 유전 알고리즘을 개선하고 있다.

첫째, 기존의 유전알고리즘은 기본적으로 제약 조건이 없는 최적화 문제에 적용되며 제약조건이 있는 최적화 문제를 해결하기 위해서는 제약조건을 비제약조건으로 만든 다음 유전알고리즘을 적용하여 해결하고 있다. 기존의 유전알고리즘에서는 제약을 비제약으로 만들기 위해서 Goldberg [7]는 아래의 식(1), (2)과 같은 외부위반함수인 위반상관계수(Penalty coefficient) r 를 사용하고 있으며, Lin[12]은 아래의 식(3), (4), (5), (6), (7)에서와 같이 위반한계함수(Bounded penalty function) r 를 사용하였다. 식 (6)에서 G 는 식(1)의 위반항이 되고, $\langle g_j \rangle^2$ 은 식(2)의 Γ 과 같은 의미이다.

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

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

$$F = f + \bar{P} \tag{3}$$

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

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

$$G = r \sum_{j=1}^p \langle g_j \rangle^2 \tag{6}$$

$$\bar{L} = pF_{av} \tag{7}$$

<기호정의>

F : 적합도 함수

f : 목적함수

r : 위반상관계수(Penalty coefficient)

g_j : 부등제약함수

p : g_j 의 개수

\bar{P} : 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 γ 에 의해서 제약조건에 접근하게 되는데 실제적인 문제에서 γ 은 명목적인 충분한 정도의 값으로 각 제약조건과 목적함수의 형태에 따라 대단히 큰 값을 선택적으로 정하고 있다. (예를들면 한용호 등[4]의 논문에서는 γ 의 값을 1000으로 주고 있다).[7]

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

따라서 본 연구에서는 제약조건을 비제약조건으로 만들기 위해 실제 Programming과정에서 제약조건을 모두 *if* 조건문으로 처리하여 제약조건을 벗어나는 개체는 재발생시켜 제약조건을 만족할때 까지 반복을 하도록 하였다.

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 되지 않는다.

따라서 제약조건이 있는 최적화 문제이더라도 상관없이 최적해를 찾도록 하였기 때문에 일정한 규칙이 없는 위반상관계수 혹은 위반한계함수 r 과 선형변환기울기 α 를 제외시켰다.

둘째, 기존의 유전알고리즘에서는 해의 탐색중에 몇개의 적합도가 높은 개체가 발생할 경우에 이들 개체들이 전체 집단에서 아주 큰 비율을 차지하게 되며 이것이 미성숙 수렴을 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 구하고자 하는 최적해의 개체가 되기 위해서는 특별히 개체에 대한 조정이 필요하며 본 연구에서는 이를 위해 적합도 변환(Fitness scaling)을 하였다. 즉 각각의 유전 연산을 거쳐 생성된 새로운 개체를 평가하여, 제약조건을 만족하지 못한다면 이 개체를 무작위로 다시 발생시켜 재평가하는 단계를 반복하였고, 제약조건을 만족한다면 적합도 변환을 하지 않고 다음 단계로 바로 진행하도록 하였다.

위와 같이 제약조건을 만족하지 못할 경우에 행하는 적합도 변환은 탐색공간에 대해 광범위한 탐색이 가능하며 따라서 해의 탐색에서 개체의 조기수렴현상을 방지하고, 개체의 다양성을 높여 수많은 세대를 거치지 않고서도 빠르게 최적해에 도달하도록 하였다.

셋째, 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 유전시킴으로서 최적해에서 멀리 떨어진 공간만을 탐색하게 되어 결국에는 최적해를 찾지 못하는 경우도 발생한다.

따라서 이러한 문제점을 개선하기 위하여 본 연구에서는 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.

<그림 2>는 본 연구에서 제시한 IGA의 구체적 절차를 흐름도를 통해 보여주고 있으며, 흐름도의 마지막 부분에서 사용한 종료조건은 반복되는 세대수를 프로그램설계상에서 미리 설정해 주는 방법을 택하고 있다.

4. 사례연구

Model 1 : Goldstein-Price 함수[1]

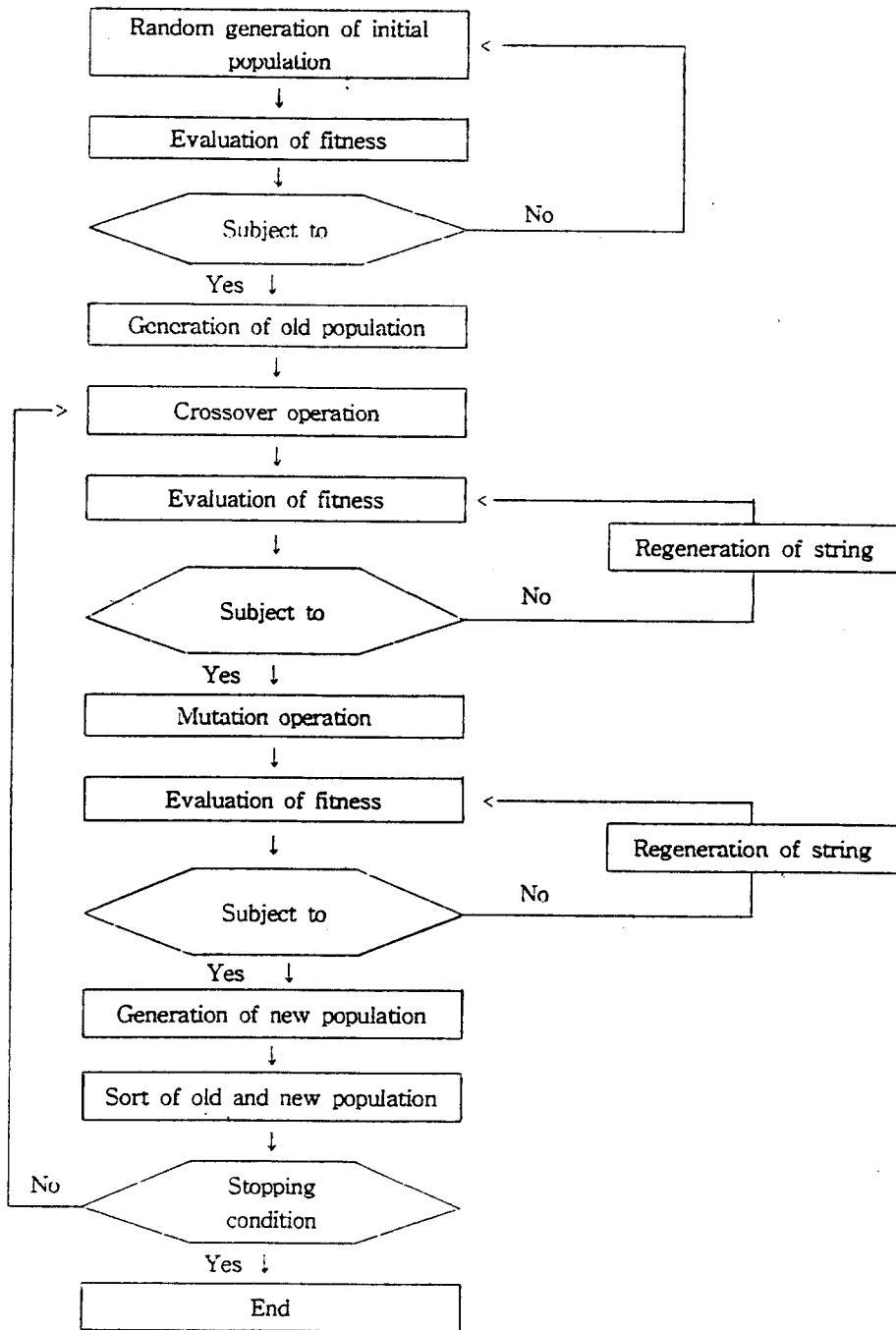
Objective function

$$\begin{aligned} \text{Minimize } F(x) = & [1+(x_1+x_2+1)^2(19-14x_1 \\ & + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \\ & \times [30+(2x_1-3x_2)^2(18-32x_1+12x_1^2 \\ & + 48x_2-36x_1x_2 + 27x_2^2)] \end{aligned}$$

Subject to $-2.0 \leq x_1 \leq 2.0$

$$-2.0 \leq x_2 \leq 2.0$$

이동곤등[3]은 위의 Model 1을 이용하여 기존의 유전알고리즘과 Hooke & Jeeves[13]의 직접탐색법(Direct search method), 이 두가지 해법을



<그림 2> 개선된 유전 알고리즘(IGA)의 흐름도

결합한 혼합모델(Hybrid model)을 제시하여 비교, 분석하였다. 따라서 본 연구에도 같은 모델을

사용하여 그 실행결과를 비교하여 <표 1>에 제시하였다.

<표 1> Optimal solution of Model 1

	Genetic Algorithm		Hooke & Jeeves Direct Search Method		Hybrid method	IGA	Type of Variables
x_1	-0.01	0.00	1.19	-0.60	0.00	0.00	continuous
x_2	-0.97	-1.00	7.99	-3.99	-1.00	-1.00	continuous
$F(x)$	3.52	3	840	30	2.99	3.00	
run time(sec.)	3.9	5.8	0.03	0.03	0.32	0.3	
population size	200	200			100	10	
generation number	20	30			10	14	
starting points			0.5, 0.5	-0.5, -0.5			
step size, time(sec)			0.3	0.3			

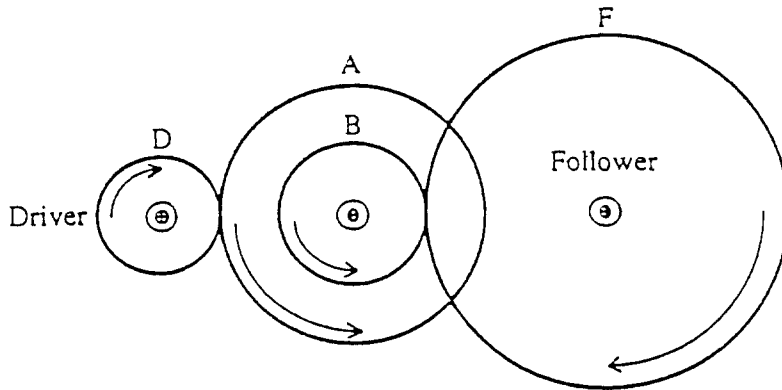
<표 1>의 결과를 살펴보면 Hooke & Jeeves의 직접탐색법의 경우, $F(x)$ 값이 각각 840과 30으로 두 경우 모두 국소최적점에 빠져 최적해를 찾지 못하였다. 하지만 본 연구에서 개발한 IGA의 경우(집단의 크기(Population size) : 10, 교차변이율(Crossover rate) : 0.4, 돌연변이율(Mutation rate) : 0.1, 세대수(Generation number) : 30) 세대수가 14번째에서 $x_1 = 0.00$ $x_2 = -1.00$ 일 때, 최적해 3.00이 0.3초만에 얻어졌는데 이것은 기존의 유전알고리즘 방법보다는 실행시간, 집단의 크기, 세대수에서 상당히 우수하며, 이동곤 등 [3]이 제시한 혼합모델과 비교하여도 거의 같은 결과를 도출하면서 집단의 크기의 경우 10배정도 더 우수하였다.

Model 2 : Design problem of a Gear Train

<그림 3>은 Sandgren[15]이 제시한 Gear Train의 구조를 보여주고 있다. 여기에서 Gear (속도)비율은 $1/6.931(=0.144279)$ 과 가능하면 같도록 유지해야 한다. Gear Train의 Gear비율은 다음과 같다.

$$Gear\ ratio = \frac{T_a T_b}{T_d T_f}$$

여기에서 T_a, T_b, T_d, T_f 는 각각 Gear A, B, D, F의 톱니의 수이다. 따라서 T_a, T_b, T_d, T_f 는 설계변수로 고려되며, 각 Gear에 대해 톱니의 수는 정수(Integer)이며, 12개에서 60개 사이의 값을 가진다. 편의상 설계변수 T_a, T_b, T_d, T_f 를



〈그림 3〉 Gear train의 구조

를 x_1, x_2, x_3, x_4 로 변환하면 다음과 같다.

$$Find X = [T_d, T_b, T_a, T_f]^T = [x_1, x_2, x_3, x_4]^T$$

수리적 모델은 다음과 같다.

Objective function

$$Minimize F(x) = \left(\frac{1}{6.931} - \frac{x_1 x_2}{x_3 x_3} \right)^2$$

$$Subject\ to\ x_i \geq 12 \quad i=1, \dots, 4$$

$$x_i \leq 60 \quad i=1, \dots, 4$$

Model 2에서 제시한 모델을 적용한 결과를 〈표 2〉에 제시하였으며, 〈표 2〉를 보면 기존의 비선형 이산 정수계획법(Nonlinear integer discrete programming)을 이용한 Sandgren[15]의 연구와 혼합 정수-이산-연속계획법(Mixed integer - discrete - continuous programming)을 이용한 Fu et al. [6]의 연구등에서 구한 최적해보다 Meta-genetic algorithm을 이용한 Wu 와 Chow[16]의 연구와 본 연구에서 개발한 IGA(집단의 크기 : 40, 교차변이율 : 0.4, 돌연변이율 : 0.15, 세대수 : 3000)가 세대수 : 527세대에서 최적해를 구하는데 더 뛰어난 성능을 보여주고 있

음을 알 수 있다. 그러나 Wu & Chow의 연구에서 제시한 Meta-genetic algorithm은 기존의 유전알고리즘에서 사용되는 모든 parameter들의 조합을 전부 사용할 경우 최적해의 parameter 집합을 못 찾을 수도 있다는 단점을 보완하여 기존의 유전알고리즘의 Genetic Parameter를 String 자체로 Code화 시킨 유전알고리즘이다. 하지만 본 연구의 IGA는 기존의 유전알고리즘에 대해 이러한 변형을 거치지 않고 단지 몇몇 모호한 개념만을 수정하여 제시한 개선된 유전알고리즘으로서 Wu와 Chow의 연구결과와 같은 최적해를 도출하였기 때문에 더 간단하고, 효율적인 기법이라고 할 수 있다.

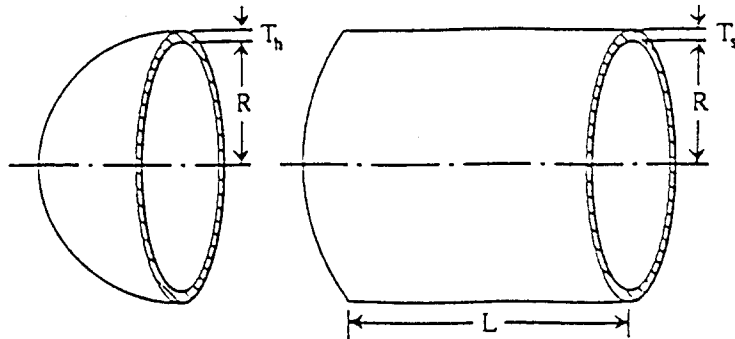
Model 3 : *Design Problem of a Pressure Vessel*

Pressure Vessel 구조는 Sandgren[15]이 제시했으며 〈그림 4〉에 나타나 있다. 설계변수는 Vessel의 사양에 필요한 치수이며, 다음과 같이 표시된다.

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

〈표 2〉 Optimal solution of Model 2

	Sandgren	Fu et al.	Wu & Chow	IGA	Type of Variables
x_1	18	14	19	19	integer
x_2	22	29	16	16	integer
x_3	45	47	43	43	integer
x_4	60	59	49	49	integer
$F(x)$	5.7×10^{-6}	4.5×10^{-6}	2.7×10^{-12}	2.7×10^{-12}	



〈그림 4〉 Pressure vessel의 구조

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

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

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

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

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

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

이상과 같은 이산·연속변수가 서로 혼합된 조합최적화 문제에 IGA를 적용하여 기존의 연구들과 비교·분석한 결과는 〈표 3〉과 같다.

〈표 3〉의 Fu et al. [6]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 [15]의 결과는 분지한계법(Branch and bound)을 이용하여 해를 구하고 있지만 적용기법의 특

〈표 3〉 Optimal solution of Model 3

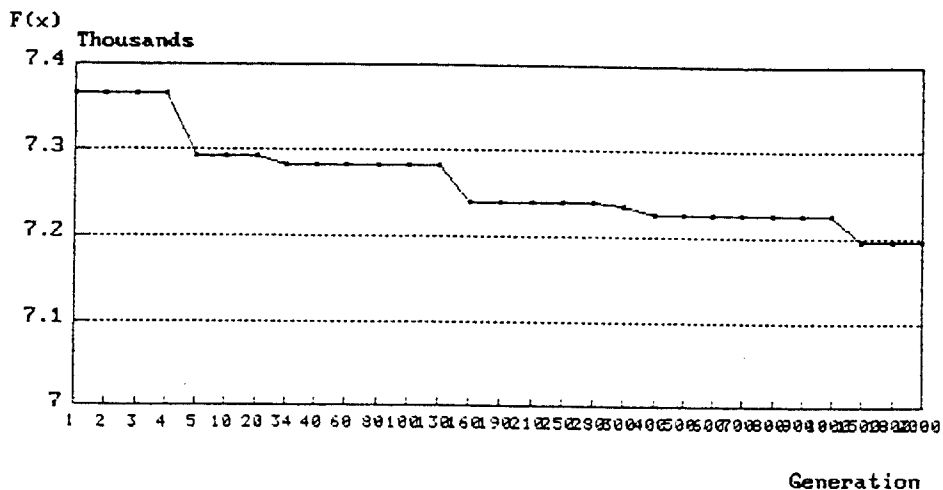
	Fu et al.	Sandgren	IGA	Type of Variables
x_1	1.125	1.125	1.125	discrete
x_2	0.625	0.625	0.625	discrete
x_3	48.3807	48.97	58.2901	continuous
x_4	111.7449	106.72	43.6930	continuous
F(X)	8048.619	7982.5	7197.73	

성상 초기상태에서는 전체 node의 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 IGA(집단의 크기 : 30, 교차변이율 : 0.9, 돌연변이율 : 0.15, 세대수 : 5000)에서는 세대수가 1721번째에서 F(x) : 7197.73으로 기존의 Fu등과 Sandgren이 구한 해보다

더 우수한 해를 구하였다. 이러한 현상은 기존의 두 연구 모두 국부해로 수렴하였지만 IGA는 해의 다양성을 실현하고 미성숙 수렴현상을 방지하기 위해 적합도 변환을 실시하였기 때문에 비교적 빠른 시간내에 수많은 세대를 거치지 않고서도 더 우수한 해를 구할 수가 있었다.

〈그림 5〉는 Model 3에 IGA를 적용하여 세대가 변함에 따라 최적해에 수렴하는 현상을 보여주고 있다.



〈그림 5〉 IGA를 이용한 Model 3의 최적해 탐색과정

5. 결론

최적화기법은 기존의 많은 연구들에서 수행되었으며 정형화된 기법이 개발되어 사용되고 있다. 하지만 실제 공학설계문제에서는 설계변수의 형태가 정수, 이산, 연속등으로 서로 혼합되어 사용되는 경우가 많다. 예를들면, 두 개의 금속판을 연결하는 리벳의 크기는 몇몇 이산적인 수치로 정해져 있으며, 사용되는 리벳의 수는 제한된 정수로 규정되어 있을 경우에, 이러한 설계변수들이 서로 혼합되어 최적해를 구하는 조합최적화 문제(Combinational optimization problem)는 아직 정형화된 기법이 도출되지 않고 있는 실정이다.

유전알고리즘은 자연계의 생물 유전법칙에 근거하고 있으며 변수의 형태(예를들면 이산, 정수, 연속변수)가 여러 가지로 조합되어 사용될 경우에 효과적인 최적해 탐색 알고리즘으로 최근에 많은 응용분야에 적용되어 좋은 결과를 도출하고 있다. 하지만 기존의 유전알고리즘에서는 제약조건이 있는 최적화 문제에 대해 외부위반함수(Exterior penalty function)를 도입하여 제약조건이 없는 최적화 문제로 변형하여 해를 구한다. 그러나 위반함수 도입시 결정해야 할 위반상관계수에 대해 확정적인 규칙이 없고, 문제에 따라 적합도 변환(Fitness scaling)을 달리 적용하여 구하고 있는 실정이다.

이러한 문제점을 해결하기 위해 본 연구에서는 제약조건 자체를 if 조건문으로 처리하여 모호한 의미의 외부위반함수 도입을 배제하였고, 각 연산마다 개체를 재평가하여 적합도 변환없이 집단의 다양성을 시도하였다. 그 결과 탐색공간에 대해 광범위한 탐색이 가능하며 해의 탐색에서 개체의 미성숙 수렴현상을 방지하고 개체의 다양성을 높여 수많은 세대를 거치지 않고서도 빠르게 최적해에 도달하도록 하였으며, 3가지 모델(Model 1

: 연속변수, Model 2 : 정수변수, Model 3 : 혼합변수)을 사례연구하여 비교·분석하였다.

Model 1의 분석에서는 본 연구에서 제시한 IGA기법이 기존의 유전알고리즘보다 실행시간, 집단의 크기, 세대수등에서 상당히 우수함이 입증되었고, Model 2의 분석에서도 기존 연구중에서 우수한 해를 도출한 Wu와 Chow[16]의 연구와 비교하여도 동일한 결과의 최적해를 얻을 수 있었다. 특히 Model 3은 설계변수의 형태가 이산변수와 연속변수가 혼합된 조합최적화 문제로 본 연구에서 제시한 IGA기법이 기존의 여러연구들 [6][15]에서 제시한 다른 알고리즘의 적용결과보다 더 뛰어난 해를 구할 수 있음을 보여주고 있다.

유전알고리즘은 주어진 문제마다 복제의 방법, 교체를, 돌연변이를, 집단의 크기, 실행하는 세대수, 정밀도 등의 조그만 변화에도 최적해를 구하는데에 아주 민감하게 그 값이 변화한다. 따라서 이러한 문제를 해결하기 위해 본 연구에서 개발된 IGA에서는 제약조건을 벗어나는 범위의 개체를 재발생시키고 있기 때문에 그 벗어나는 범위를 어느정도로 하느냐에 따라 재발생시키는 빈도가 차이가 나게 되며, 이것이 개체의 다양성에 영향을 주게 됨으로 유전알고리즘에 사용되는 모든 값들을 서로 조합적으로 연관시켜 실행할 수 있는 체계를 구축하는 것이 차후의 연구과제라 할 수 있다.

참 고 문 헌

- [1] 김기화, "Genetic Algorithm에 의한 다목적 함수 최적구조설계," 서울대학교 조선해양공학과 박사학위 논문, 1993. 12.
- [2] 김창은, 주용준, "Application of Genetic Algorithm for Simple Parameter Optimiza-

- tion Problem,” 추계 대한산업공학회 학술 발표대회, 1994. 9, pp. 27-34.
- [3] 이동곤, 김수영, 이창역, “최적화의 효율향상을 위한 유전해법과 직접탐색법의 혼용에 관한 연구,” 「산업공학」, Vol. 8, No. 1, 1995. 3. pp. 23-30.
- [4] 한용호, 류광렬, “기계-부품군 형성문제의 사례를 통한 유전 알고리즘의 최적화 문제의 응용,” 「경영과학」, Vol. 12, No. 2, 1995. 8, pp. 105-127.
- [5] 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, 1993*, pp. 375-382.
- [6] 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, No. 4, 1991, pp. 263-280.
- [7] Goldberg, D. E., “Genetic Algorithms in Search, Optimization, and Machine Learning, Reading,” Massachusetts, Addison-Wesley, 1989.
- [8] Grefenstette, J. J., “Optimization of Control Parameter for Genetic Algorithm,” *IEEE Transactions of Systems, Man and Cybernetics*, Vol. 16, No. 1, 1986.
- [9] 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,” *International Journal of Computer Integrated Manufacturing*, Vol. 8, No. 2, 1995, pp. 92-101.
- [10] Holland, J. H., *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan, The University of Michigan Press, 1975.
- [11] Hon, K. K. B. and Chi, H., “A New Approach of Group Technology Part Families Optimization,” *Annals of the CIRP*, Vol. 43, 1994. 1.
- [12] Lin, C. Y. and Hajela P., “Genetic Algorithms in Optimization Problems with Discrete and Integer Design Variables,” *Engineering Optimization*, Vol. 19, 1992, pp. 309-327.
- [13] Person M. G., “Optimization Method for use in Computer-Aided Ship Design,” *STAR Symposium, SNAME*, 1975, pp. 13. 1- 13. 27.
- [14] Rajeev, S. and Krishnambros, C. S., “Discrete Optimization of Structures Using Genetic Algorithms,” *ASCE, Journal of Structural Engineering*, Vol. 118, No. 5, 1992, pp. 1233-1250.
- [15] Sandgren, E., “Nonlinear Integer and Discrete Programming in Mechanical Design Optimization,” *Transactions of the ASME, Journal of Mechanical Design*, Vol. 112, No. 2, 1990, pp. 223-229.
- [16] Wu, S. J. and Chow, P. T., “Genetic Algorithms for Nonlinear Mixed Discrete-Integer Optimization Problems via Meta-Genetic Parameter Optimization,” *Engineering Optimization*, Vol. 24, 1995, pp. 137-159.