⊠ 응용논문

유전알고리즘을 이용한 최적생산설계
- Optimal Production Design Using Genetic Algorithms -

류 영 근*
Ryku, Young Keun*

Abstract

An optimization problem is to select the best of many possible design alternatives in a complex design space.

Genetic algorithms, one of the numerous techniques to search optimal solution, have been sucessfully applied to various problems (for example, parameter tuning in expert systems, structural systems with a mix of continuous, integer and discrete design variables) that could not have been readily solved with more conventional computational technique. But, conventional genetic algorithms are ill defined for two classes of problems, ie., penalty function and fitness scaling.

Therefore, this paper develops Improved genetic algorithms(IGA) to solve these problems. As a case study, numerical examples are demonstrated to show the effectiveness of the Improved genetic algorithms.

1. 서론

기계설계최적화 기법은 기존의 많은 연구들을 통해서 개발되어 왔으며 그 해법 또한 다양하게 적용되고 있다. 즉 여러가자 제약조건들을 모두 만족해야 하는 설계변수들의 집합중에서 최적해를 구하는 방법에 대하여서는 많이 연구되어 왔으나 아직 설계변수를 직접 사용하기에는 해결해야 할 문제가 많이 남아 있는 상태이다. 예를들면 설계변수들이 정수(Integer), 실수(Real number) 또는 이산적(Discrete) 수치를 함께 가지며 서로 혼합되어 사용되는 혼합형 최적화 문제(Combinational optimal solution)와 여러개의 지역적 최적점(Local optimum)이 존재하는 경우에 전범위 최적점(Global optimum)을 효율적으로 구하는 문제등을 들 수 있다.

이와 같은 형태의 문제에 대해 기존의 어떠한 최적화 기법을 적용하여 해결하였다 하더라도 이를 변형하지 않고 다른 문제에 직접 적용하기에 어려운 경우가 대부분이며, 적용 자체가 불가능한 경우도 종종 발생한다. 그러므로 기존의 최적화 기법을 사용하여 이를 해결하려면 먼저 적절한 최적방법을 선택하여야 하며 해의 최적점 도달을 위해 변수설계의 많은 부분을 수정해야 한다는 번거로운 과정을 거쳐야만 했었다. 대표적인 예로서 기존의 최적화 기법인 선형계획법 및 휴리스틱(Heuristic)기법등은 해를 발생할 공간이 1차원문제인 경우에는 최적점을 쉽게 찾을 수 있지만 2차원이상의 복잡한 문제(예를들면, 비선형계획법(Non-Linear programming),

† 이 논문은 1997년도 안동과학대학 학술지원금으로 수행되었슴.
* 안동과학대학 산업정보공학과

다목적 계획법 (Multi-objective programming))등에서는 최적해를 도출하는 계산절차가 복잡하고 그 적용범위가 좁다는 단점이 있다.

이러한 문제점을 비교적 쉽게 해결하여 줄 수 있는 인공지능(Artificial Intelligence)의 한 기법인 유전해법(Genetic Algorithm)은 생태계의 진화원리에 바탕을 둔 최적해에 대한 탐색 알고리즘으로 다양한 종류의 문제들에 효율적으로 적용가능하며 비교적 전범위의 최적점에 근사한 해를 구할수 있다는 장점이 있다. 하지만 유전 해법 설계시 적용되는 여러 변수에 대한 정형화된 기법이 아직까지 없는 상태이며, 문제의 유형에 따라 설계자가 변수들에 입력되는 수치를 임의로 변경하여 이를 여러번 반복 수행하여 최적해를 도출하고 있는 실정이다. 즉 아직까지 정형화된 유전해법이 도출되고 있지 않은 상태이다.

따라서 본 연구에서는 최적화 문제에 대한 유전해법의 가치를 평가해 보기 위해 기존의 유전해법 적용시에 문제가 되고 있는 위반함수(Penalty function) 적용문제와 미성숙 수렴(Premature convergence)현상을 방지하기 위해 적용하는 적합도 변환(Fitness scaling)방법, 우수개체의 탈락현상등을 개선시킨 새로운 유전해법을 제시하였으며, 이를 두가지 최적화문제에 적용하여 본 연구에서 제시된 해법의 우수성을 입증하고자 한다.

2. 유전 해법

유전해법은 자연선택(Natural selection)과 적자생존(Survival of the fittest)에 근거를 두고 있으며, 새로운 집단(New population)을 형성할 때에 이전 집단(Old popuation)에서 높은 적합도값을 가지는 개체(String)가 하나 또는 그 이상의 후손(Offspring)을 만드는데 더 높은 확률을 가지고 유전된다는 것이 그 기본적인 원리이다.[9]

유전해법은 Holland [8]가 개발하였으며, Goldberg [5]에 의해 가스 송수관에 대한 최적 설계가 최초로 시도된 이래 많은 발전이 있었다.

유전해법에 대한 기존의 연구를 살펴보면, Lin과 Hajela[11]는 이산설계변수와 정수설계변수가 혼합된 최적설계문제(Mixed discrete-integer optimal design problem)에 적합한 유전 해법을 개발하였으며, Rajeev와 Krishnambros[13]는 이산화변수(Discrete sizing variable)를 이용하여 구조적 최적화 문제(Structural optimization problem)에 대한 유전해법의 적용을 연구하였다. Grefenstette[6]는 유전해법에 의한 최적설계시에 사용되는 설계변수의 여러가지 조합에 대해 몇몇 수학적 함수를 대상으로 최적화를 행하여 그 활용도에 관한 연구를 하였고, Gupta등[7]은 GT(Group technology)에서는 셀제조(Cellular manufacturing)의 설계에 유전해법을 적용하였고, Fang 등[3]은 Job Shop Scheduling의 최적화문제에 대하여 유전해법의 적용사례를 발표하였다.

3. 기존 유전해법의 문제점과 개선된 유전해법(IGA)의 설계

본 연구는 크게 3가지 측면에서 기존 유전해법의 문제점을 지적하고 이를 개선한 유전해법을 제시하였다.

첫째, 기존의 유전해법에서는 제약조건이 있는 최적화 문제(식 (1))에 대해 제약조건을 비제약조건으로 만들기 위해서 Zhang과 Wang[16]은 식 (2)와 같은 위반제약식(ϕ)과 위반상관계수(δ)를 사용하였다.

$$\begin{array}{lll} \textit{Minimize} & f(x) & \\ \textit{Subject to} & g_i(x) \geq 0 & i=1,2,\ldots m \\ & & \textit{where } x \textit{ is an } m \textit{ vector} \end{array} \tag{1}$$

$$Minimize \quad f(x) + \varepsilon \cdot \delta \sum_{i=1}^{p} \psi_i \tag{2}$$

where p is the total number of constraints
δ *is a penalty coefficient*
ε *is* -1 *for maximization problems*
$+1$ *for minimization problems*
ψ *is a penalty related to the* $i-th$
constraints $(i = 1, 2, .., p)$

위의 식(1)에서 제약조건을 비제약조건으로 만드는데 사용된 위반제약식(ψ)과 위반상관계수(δ)은 프로그램 설계과정에서 제약조건을 위반하는 개체에 대해 적용하는 것으로 제약조건과 목적함수의 형태에 따라 프로그램 설계자가 임의로 정하고 있는 실정이다. 이러한 설계방식은 유전해법이 광범위하게 적용하는데 큰 문제점으로 지적되고 있다.

하지만 본 연구에서는 이러한 모호한 의미의 위반제약식과 위반상관계수를 프로그램 설계시에 제외시키며 제약조건 자체를 모두 *if*조건문으로 처리하여 이를 해결하고 있다. 만약 제약조건을 벗어나는 개체(*if*조건문을 만족시키지 못하는 개체)에 대해서는 제약조건을 만족할 때까지 개체를 재발생시키도록 하였다. 따라서 프로그램설계시 제약조건의 유무에 관계없이 해를 탐색하도록 하였다.

둘째, 기존의 유전해법에서는 새로운 개체가 발생될 때 집단중의 여러개체들을 적합도에 따라 다음세대로의 유전여부를 결정하게 된다. 만일, 초기에 몇세대를 거치지 않고서 아주 우수한 개체가 발생될 경우 이들 개체들은 다른 개체에 비해 절대적인 적합도 우위를 갖게 되어 세대가 계속 진행되더라도 더 이상의 개선은 기대할 수 없는 결과를 가져온다.

이러한 문제점을 개선하기 위하여 기존연구들에서는 적합도 변환(Fitness scaling)방법을 사용하고 있지만 이 방법 역시 적용되는 모수들을 설계자가 임의로 정하고 있다는 단점이 있다.

따라서 본 연구에서는 이러한 적합도 변환방법을 사용하지 않고 각각의 유전연산을 거쳐 새로운 개체를 평가하여 제약조건을 만족하지 못한다면 이 개체를 무작위로 다시 발생시켜 재평가하며, 제약조건을 만족한다면 적합도 변환을 하지 않고 다음 단계로 바로 진행하도록 하였다.

위와 같이 제약조건을 만족하지 못할 경우에 행하는 방법은 탐색공간에 대해 광범위한 탐색이 가능하며 따라서 해의 탐색에서 개체의 조기수렴현상을 방지하고, 개체의 다양성을 높여 수많은 세대를 거치지 않고서도 빠르게 최적해에 도달할 수 있다는 장점이 있다.

셋째, 기존의 유전해법은 확률적인 선택을 하기 때문에 다음세대의 자손을 생성하는데 있어 이전집단에 높은 적합도를 지닌 개체가 존재하더라도 다음세대에서 랜덤하게 개체가 평가되어 새로운 개체가 선택됨으로 이전집단에서 구한 우수개체가 다음 자손세대에 선택되지 않을 수도 있다. 반면에 우수하지 않은 개체들을 다음 세대로 계속하여 유전시킬 경우에는 최적해에서 멀리 떨어진 공간만을 탐색하게 되어 결국에는 지역적 최적점에 수렴하여 더 이상의 개선점을 찾지 못하는 경우도 발생한다.

따라서 이러한 문제점을 해결하기 위하여 IGA에서는 이전집단에서 최고의 적합도를 갖는 하나의 개체를 강제적으로 다음 세대의 자손집단에서 선택되도록 하고 이전집단의 개체와 다음 세대의 자손집단에서 선택된 개체 전부를 정렬(Sorting)하여 높은 적합도를 가지는 순으로 집단의 크기 만큼 다음세대를 선택하여 새로운 자손세대를 만드는 방법을 사용하였다.

본연구에서 사용되는 유전연산자인 선택(Selection)방법은 기존의 엘리트(Elitism)전략을 수정하여 사용하였고, Crossover 연산자와 Mutation 연산자는 기존의 방법인 1점(one-point)연산[11]과 랜덤발생법[5]을 사용하였다.

<그림 1>은 본 연구에서 제시한 IGA의 계산과정을 보여주고 있으며, <그림 1>의 마지막 부분에서 사용한 종료조건(Termination condition)은 반복되는 세대수를 프로그램 설계자가 미리 설정해 주는 방법을 택하고 있다.

```
Improvement Genetic Algorithm(IGA)
{
    for( ; ; )
    {
        initialize population;
        evaluate population;
        for( ; ; )
        {
            for( j=1,2,…,p )
            {   if ( not g_j )    recreate string;
                else break;
            }
        }
        reproduction among population;
        crossover parents;
        for( ; ; )
        {
            for( j=1,2,…,p )
            {   if ( not g_j )    recreate string;
                else break;
            }
        }
        mutate parents;
        for( ; ; )
        {
            for( j=1,2,…,p )
            {   if ( not g_j )    recreate string;
                else break;
            }
        }
        generate new_population
        sort population and new_population;
        population = new_population;
        if ( termination condition ) break;
    }
}
```

<그림 1> 개선된 유전해법(IGA)의 계산과정

4. 사례연구

사례연구에서는 기계설계최적화 모델에서 많이 사용되는 기어트레인 최적설계(Optimal design of gear train)모델과 압력용기 최적설계(Optimal design of pressure vessel)모델을 제시하였으며 이들 두 모델은 기존에 개발된 최적화 기법들을 이용해 그 해가 이미 구해졌기 때문에 본 연구에서 제시한 IGA와의 비교·분석이 가능하다.

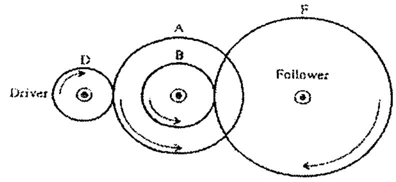
본 연구에서 개발한 IGA는 IBM 호환기종 컴퓨터(Pentium-166MHz, RAM : 32M)에서 Visual Basic을 이용하여 프로그래밍(Programming)하였다.

4.1 사례모델 1

<그림 2>는 Gear Train의 구조를 보여주고 있다. 이 모델은 Sandgren[14]이 제시한 것으로 여기에서 Gear 속도 비율은 식(3)과 가능하면 같도록 유지해야 최적의 효과를 볼 수 있으며 Gear Train의 Gear 비율은 식 (4)와 같이 표시가능하다.

$$\frac{1}{6.931} = 0.144279 \tag{3}$$

$$Gear\ ratio = \frac{T_d T_b}{T_a T_f} \tag{4}$$



<그림 2> 기어 트레인의 구조

여기에서 T_a, T_b, T_c, T_f는 각각 Gear A, B, C, F의 톱니의 수이다. 따라서 T_a, T_b, T_c, T_f는 설계변수로 고려되며, 각 Gear에 대해 톱니의 수는 설계상 정수(Integer)이며, 12개에서 60개 사이의 값을 가진다. 설계변수 T_d, T_b, T_a, T_f 를 표현의 편의상 x_1, x_2, x_3, x_4로 변환한후 수리적 모델을 나타내면 식(5)와 같다.

$$Minimize \quad F(x) = \left(\frac{1}{6.931} - \frac{x_1 x_2}{x_3 x_4} \right)^2 \tag{5}$$

$$\begin{aligned} Subject\ to \quad & x_i \geq 12 \quad & i = 1, \ldots, 4 \\ & x_i \leq 60 \quad & i = 1, \ldots, 4 \end{aligned}$$

사례모델 1은 4개의 정수설계변수로 이루어진 비선형 최소화문제로 기존의 연구들과 IGA의 적용결과를 <표 1>에 제시하였다.

<표 1> 사례모델 1의 적용 결과

Methods	MINSLIP	IDCNLP	B&B	SA	GA	IGA
x_1	19	14	18	30	19	19
x_2	16	29	22	15	16	16
x_3	42	47	45	52	43	43
x_4	50	59	60	60	49	49
F(x)	2.33×10^{-7}	4.5×10^{-6}	5.7×10^{-6}	2.36×10^{-8}	2.7×10^{-12}	2.7×10^{-12}

<표 1>를 살펴보면 Loh등[12]의 연구는 MINSLIP(Mixed Integer Nonlinear Sequential Linearization Programming Algorithm)를 이용하였으나 비선형문제를 선형화할때 실제의 정확한 최적점의 값을 잘라버리는 결과를 초래하여 지역적 최적점에 빠져 버리는 단점이 있다.

Fu등[4]의 적용결과는 IDCNLP(Integer-discrete-continuous non-linear programming) 알고리즘을 이용하여 해결하였는데, 이 방법은 일반적인 비선형계획법의 해결절차에 근거를 두고 있으며 문제 해결과정에서 초기입력변수의 값을 선택하는데 설계자의 주관적 판단에 의해 결정하기 때문에 올바른 입력변수값을 정하지 못한다는 단점이 있다.

Sandgren[14]의 연구는 Branch and Bound법을 이용하였으나 적용기법의 특성상 문제의 규모가 복잡할때는 해결하기가 곤란하다는 단점이 있다.

Zhang등[16]의 연구는 SA(Simulated Annealing)을 이용하였지만 기존연구들과 마찬가지로 지역적 최적점에 빠져버렸다. 이러한 현상은 탐색과정에서 더 나은 해를 찾기 위하여 탐색공간내의 지역적 최적점 근처의 탐색(neighbornood search)을 더 이상 하지 않았기 때문인 것으로 분석된다.

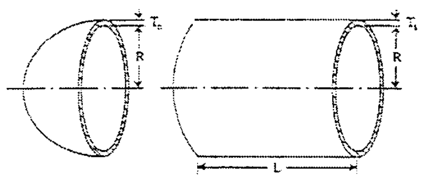
그러나 Wu 와 Chow[15]의 연구에서 제시한 메타 유전해법(Meta-genetic algorithm)은 기존의 유전해법에서 사용되는 모든 모수들의 조합을 전부 사용할 경우 최적해의 모수집합을 찾지 못

한다는 단점을 보안하여 기존의 유전해법의 유전모수(Genetic Parameter)를 개체자체로 코드(Code)화 시킨 유전해법이다. 하지만 본 연구의 IGA는 집단의 크기 : 25, 교차변이율 : 0.4, 돌연변이율 : 0.1, 세대수 : 2000)으로 했을 때 세대수가 754세대에서 수렴하였으며 이 결과는 Wu와 Chow의 연구결과를 제외하고는 기존의 연구들보다 더 뛰어난 성능을 보여주고 있음을 알 수 있다.

따라서 IGA는 기존의 유전해법에 대해 Wu 등이 제시한 변형을 거치지 않고 단지 몇몇 모호한 개념만을 수정하여 제시한 개선된 유전해법으로서 Wu와 Chow의 연구결과와 같은 최적해를 도출하였기 때문에 더 간단하고, 효율적인 기법이라고 할 수 있다.

4.2 사례모델 2

압력용기(Pressure Vessel)는 운전중에 일어나는 압력 및 온도를 기준으로 설계하는데 그 구조는 Sandgren[14]이 제시했으며 <그림 3>에 나타나 있다.



<그림 3> 압력용기의 구조

설계변수는 용기의 제작사양에 필요한 치수이며, 다음과 같이 표시된다.

$$\begin{aligned} Find \quad X &= [\, T_s,\ T_h,\ R,\ L \,]^T \\ &= [\, x_1,\ x_2,\ x_3,\ x_4 \,]^T \end{aligned}$$

수리적 모델은 아래와 같다.

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

Subject to

$$\begin{aligned} G_1(X) &= x_4 - 240 \le 0 \\ G_2(X) &= x_1 - 1.1 \ge 0 \\ G_3(X) &= x_1 - 0.0193 x_3 \ge 0 \\ G_4(X) &= x_2 - 0.00954 x_3 \ge 0 \\ G_5(X) &= x_2 - 0.6 \ge 0 \\ G_6(X) &= \pi x_3^2 x_4 + \frac{4}{3}\pi x_3^3 - 750 \times 1728 \ge 0 \end{aligned}$$

이 모델에서는 모두 4개의 설계변수가 사용되는 비선형 최소화 문제로 설계변수 x_1, x_2는 이산설계변수이며, 0.0625 정수배수(Integer multiples)를 가진다. x_3, x_4는 연속설계변수이며 목적함수는 압력용기의 총제조비용을 최소화하는 것이다.

이상과 같은 이산·연속설계변수가 서로 혼합된 혼합최적화 문제에 대해 본연구에서 개발한 IGA를 적용하여 기존의 연구들과 비교·분석한 것이 <표 2>에 제시되어 있다.

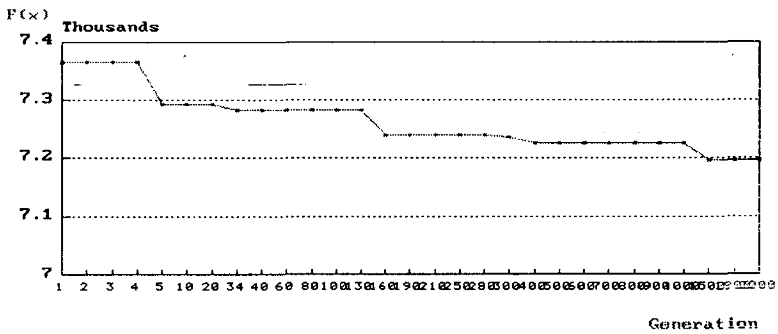
<표 2> 사례모델 2 의 적용결과

Methods	IDCNLP	B&B	GA	IGA
$x_1(Ts)$ (inch)	1.125	1.125	1.125	1.125
$x_2(Th)$ (inch)	0.625	0.625	0.625	0.625
$x_3(R)$ (inch)	48.3807	48.97	58.1978	58.2901
$x_4(L)$ (inch)	111.7449	106.72	44.2930	43.6930
$g_1(x)$	-0.191250	-0.179	-0.001782	-0.00000107
$g_1(x)$	-0.163449	-0.1578	-0.069793	-0.068912446
$g_1(x)$	-75.8750	-3.0	-974.5829	-0.41506
$g_1(x)$	-128.2551	-133.284	-195.7070	-196.307
$F(X)$	8048.619	7982.5	7207.49	7197.03

<표 2>의 IDCNLP(Integer - discrete - continuous non-linear programming) 알고리즘은 Fu등[4]이 제시한 결과로 이 방법은 사례모델 1의 경우와 마찬가지로 초기입력변수의 값을 선정하는데 어려움이 있어 지역적 최적점에 수렴하였으며, Sandgren[14]의 연구는 Branch and bound을 이용하여 해를 구하고 있지만 적용기법의 특성상 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. Wu 등[15]의 연구결과는 메타유전해법을 이용하였지만 기존의 연구들과 마찬가지로 지역적 최적점에 빠져 더 우수한 해를 구하지 못하고 있다.

반면에 본 연구의 IGA(집단의 크기 : 25, 교차변이율 : 0.4, 돌연변이율 : 0.15, 세대수 : 4300)에서는 세대수가 1710번째에서 목적함수값이 7197.03으로 기존의 연구들보다 더 우수한 해를 구하였다. 이러한 현상은 기존의 두 연구 모두 지역해로 수렴하였지만 IGA는 해의 다양성을 실현하고 미성숙 수렴현상을 방지하였기 때문에 수많은 세대를 거치지 않고서도 더 우수한 해를 구할 수 있었다고 분석된다.

<그림 4>는 사례모델 2에 대해 IGA를 적용하여 해의 수렴현상을 보여 주고 있다.



<그림 4> IGA를 이용한 사례모델 2의 수렴과정

5 결론

본 연구에서는 기계설계 최적화 문제를 해결하기 위한 개선된 유전해법의 개발에 그 목적이 있다. 기계설계 최적화문제는 정수,이산, 혹은 실수설계변수가 혼합되어 사용되며 설계상 정밀

한 설계를 요구하기 때문에 문제의 형태가 비선형인 경우가 대부분이다. 이러한 비선형 혼합설계변수가 사용된 기계설계 최적화문제는 기존의 연구들을 통해 해결되어 왔지만 문제의 형태와 규모에 따라 달리 적용해야 된다는 단점이 있다.

하지만 본 연구에서 제시한 새로운 유전해법은 기존 유전해법 적용시 문제점으로 지적된 몇몇 모호한 의미를 개선한 것으로 문제의 규모나 형태에 관계없이 적용가능하다는 장점이 있다. 이를 증명하기 위해 두가지의 기계설계 최적화문제에 적용하여 그 유효성을 보였다.

유전해법은 사용되는 각종의 모수들(복제의 방법, 교체율, 돌연변이율, 집단의 크기, 실행하는 세대수, 정밀도등)에 따라 그 값들의 조그만 변화에도 해를 구하는데에 아주 민감하게 변화한다. 그리고, 개선된 유전 해법에서는 제약조건을 벗어나는 범위의 개체를 재발생시키고 있기 때문에 그 벗어나는 범위를 어느정도로 하느냐에 따라 재발생시키는 빈도가 차이가 나게 되며 이것이 개체의 다양성에 영향을 주게된다.

이러한 이유로 유전해법 수행에 필요한 모수들의 최적조합을 연구하는 것도 필요하다 하겠다.

참 고 문 헌

[1] 김기화, “Genetic Algorithm에 의한 다목적함수 최적구조설계,” 서울대학교 조선해양공학과 박사학위 논문, 1993.

[2] 한용호, 류광렬, “기계-부품군 형성문제의 사례를 통한 유전 해법의 최적화 문제에의 응용,” 경영과학, Vol. 12, No. 2, 1995. 8.

[3] 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.

[4] Fu, J. F., Fenton, R. G., and Cleghorn, W. L., "A Mixed Integer-Discrete -Continuous Programming Method and Its Applications to Engineering Design Optimization," Engineering Optimization, Vol. 17, pp. 263-280, 1991.

[5] Goldberg, D. E., Genetic Algorithms in Search, Optimization, and Machine Learning, Reading, Massachusetts, Addison-Wesley, 1989.

[6] Grefenstette, J. J., "Optimization of control parameter for genetic algorithm," IEEE Transactions of systems, man and cybernetics, Vol. 16, No. 1, 1986.

[7] Gupta, Y. P., Gupta, M. C., Kumar, A. K. and Sundram, C., "Minimizing total intercell and intracell moves in cellular manufacturing: a genetic alforithm approach," INT. J. Computer Integrated Manufacturing, Vol. 8, No. 2, pp. 92-101, 1995

[8] Holland, J. H., Adaptatiion in Natural and Artificial Systems. Ann Arbor, Michigan, The University of Michigan Press, 1975.

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

[10] Liepins, G. E. and Hilliard, M. R., "Genetic Algorithms : Foundations and Applications," Annals of Operations Research, Vol. 21, pp. 31-58, 1989.

[11] 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.

[12] Loh, H. T., and Papalambros, P. Y., "A Sequential Linearization Spproach for Solving Mixed - Discrete Nonlinear Design Optimization Problems," Journal of Mechanical Design," Vol. 113, pp. 325-334, 1991.

[13] Rajeev, S. and Krishnambros, C. S., "Discrete optimization of structures using genetic algorithms," ASCE, Journal of Structural Engineering, Vol. 118, No. 5, pp. 1233-1250, 1992.

[14] Sandgren, E., "Nonlinear Integer and Discrete Programming in Mechanical Design Optimization," ASME Journal of Mechanical Design, Vol. 112, No. 2, pp. 223-229, 1990.

[15] Wu, S. J. and Chow, P. T., "Genetic Algorithms for Nonlinear Mixed Discrete - Integer Optimization Problems via Meta-Genetic Paremater Optimization," Eng. Opt., Vol. 24, pp. 137-159, 1995.

[16] Zhang, C., and Wang, H. P., "Mixed-Discrete Nonlinear Optimization with Simulated Annealing," Engineering Optimization, Vol. 21, pp. 277-291, 1993.