

효율적인 그레이팅 배치를 위한 유전 알고리즘의 적용

이정규¹ · 조대호^{1†}

An Application of Genetic Algorithm for Efficient Grating Allocation

Jung Gyu Lee · Tae Ho Cho

ABSTRACT

D/In modern production industries, computer aided systems have been improving the efficiency and convenience of the various stages of work. However, as the complexity of computerized production systems increases, various techniques are still necessary. The problem we addressed occurs in computer systems that automatically make manufacturing process plans in the metal grating manufacturing industry. In the grating layout design, the key to saving the manufacturing cost is to find a design with the minimal number of cutting operations. The proposed genetic algorithm explores the feasible alternatives within the space until an optimal solution is obtained.

Key words : CAD/CAM, Genetic Algorithm, Optimization, Grating

요약

현대의 산업공정에서 CAD(Computer-Aided Design)는 설계 및 생산에 관련된 다양한 분야에서 빠져서는 안 될 중요한 분야로 자리매김 하였다. 그러나 생산 설계의 복잡도가 증가 할수록, 그와 관련된 다양한 공학 기술들이 필요로 하게 되었다. 이 논문에서는 CAD기반의 그레이팅 설계 자동화 시스템에서 분할을 고려하는 배치에 대하여 다룬다. 그레이팅 설계 디자인에서 생산 비용 절감의 핵심은 분할을 최소화 하는 디자인을 설계하는 것이다. 이를 해결하기 위하여 그레이팅 분할 배치 문제에 유전 알고리즘을 적용하여 근사 해를 탐색한다. 각 제약 조건들에 우선순위를 두어 적합도를 측정하고, 분할 선택들을 해 집합으로 구성하여 교배와 선택 연산을 수행함으로써 해를 구하고 시물레이션을 통해서 제안된 방법의 효율성을 평가한다.

주요어 : CAD/CAM, 유전 알고리즘, 최적화, 그레이팅

1. 서론

제품의 설계 및 생산과정에서 빠른 처리속도와 효율성을 제공하는 CAD(Computer-Aided-Design) 시스템에서 최적설계를 얻기 위한 방법으로 최적 해를 탐색하는 여러 알고리즘이 개발되어 사용되고 있다.^[1] 이에 따라 CAD 시스템을 사용하는 업체들은 보다 최적의 설계를 얻기 위해 공학적 분석을 통하여 각자의 생산 공정에 맞게 비용을 낮추고 효율성을 높일 수 있는 방안을 고려하였다.^[2] 자동

설계 과정에서의 탐색 문제는 가장 만족스러운 디자인이 얻어질 때까지 탐색 공간 안에서 해를 찾는 일반적으로 정형화된 탐색기법으로 볼 수 있다.^[3]

그레이팅 설계 디자인은 그레이팅의 중요한 특징의 대부분을 결정하기 때문에 설계 과정 중에서 가장 중요한 부분이다. 여기서의 결정들은 뒤에 따라오는 설계 과정 및 제조공정의 모든 영역에 영향을 미치게 된다. 그레이팅의 배치 디자인은 오류가 빈번할 뿐만 아니라 시간이 많이 걸리는 작업으로 실제 그레이팅을 생산하는 S 업체에서는 수동으로 행해져 왔다. 실제로 상당히 큰 규모의 프로젝트에서는 수천 개의 그레이팅이 제조, 배치되고 설계 자동화를 거치지 않을 경우 매우 많은 시간을 소모하게 된다. 문제의 복잡성 때문에 숙련된 디자이너가 디자인을 생성하고 가능한 기한 내의 최적의 디자인을 선정하는 것은 불가능하다. 또한 그레이팅은 마땅한 설비 시설

* 이 연구에 참여한 연구자(의 일부)는 2단계 BK21 사업의 지원비를 받았음.

2006년 7월 14일 접수, 2006년 12월 8일 채택

¹⁾ 성균관대학교 정보통신공학부

주 저 자 : 이정규

교신저자 : 조대호

E-mail; taecho@ece.skku.ac.kr

이 없는 기업에 의해 자주 디자인되며, 디자인 당시에 그레이팅에 대한 지식이 없는 상태인 경우가 많다. 따라서 가능한 디자인 대안을 생성할 수 있고 그러한 설계 자동화 능력을 가진 시스템이 요구된다.

GDS(Grating automatic Drawing System)는 실제 그레이팅을 생산하는 한 업체에서 사용하는 AutoCAD 기반의 설계 자동화 시스템이다. 설계자가 입력한 정보를 받아 작성된 도면에 맞춰 범위에 알맞게 자동으로 그레이팅을 배치하고, 세부 도면을 생성한다. 이 논문에서는 금속 그레이팅 설계 자동화 시스템에 생기는 문제를 해결하기 위해 자연 진화 방식으로 잘 알려진 유전 알고리즘을 적용하였다. 그레이팅 배치 시에 도면상에서 그레이팅 배치에 방해가 되는 개체를 오픈이라고 한다. 오픈이 존재하면 오픈에 맞추어 분할을 해야 하고 많은 비용을 소모하게 된다. 따라서 이를 최적으로 나누어 배치해야 하는 문제가 발생한다. 오픈의 수가 증가 할수록 탐색 공간은 커지게 되며, 어느 단계에 이르러서는 수동으로 계산해서는 최적 해에 접근 할 수 없을 만큼 많은 탐색 시간을 필요로 한다.

유한한 구획 내에서 특정 개체를 효율적으로 배치하는 문제를 해결하기 위하여 시뮬레이티드 어닐링, 유전자 알고리즘, 신경 회로망 등 확률론적인 방법을 사용하는 연구가 이뤄지고 있다.^[4] 이러한 방법들은 연산 수행 시간이 길다는 단점과 최적 해를 보장하지는 못하는 단점에도 불구하고 지역 해를 벗어나 최적 해에 가까운 근사 해를 보장한다는 점에서 선호되고 있다. 시뮬레이티드 어닐링은 대표적 반복 휴리스틱 알고리즘으로 주어진 함수의 전역 해(global solution)에 대한 근사치를 찾는 전역 최적화 문제의 접근 방식으로 사용된다. VLSI 최적 설계에 응용

되어 좋은 결과를 보여주었지만 온도에 따른 연산 수행 시간 증가의 위험이 있다. 유전 알고리즘은 폭넓은 적용 가능성으로 많은 배치 문제에 이용되고 있으며 분할과 관련하여 최적의 분할 경로를 찾는 연구도 이루어졌다.^[5] 기존의 유전 알고리즘을 개선하여 2차원 절단에서의 절단 경로 문제를 해결하는 해법을 제시하였다.

본 논문에서는 일반적인 분할 외에 오픈이 있는 공간에서의 분할 배치를 고려하여 알고리즘을 설계 하였으며 그에 따른 제약 조건에 만족하는 해를 찾고자 한다.

문제에 적용할 유전 알고리즘은 큰 탐색 공간 안에서 최적해(best solution)가 요구 되지 않을 때, 효과적으로 사용된다.^[6] 분할로 나누어지는 구획을 유전자로 표현하고, 선택, 교배 및 돌연변이에 의해 진화되며, 정의된 함수에 의해 유전자의 적합성을 평가한다. 평가가 끝난 후에는 적용 알고리즘이 충분히 효율적임을 시뮬레이션 결과를 통해 보인다.

본 논문의 구성은 다음과 같다. 2장에서 알고리즘이 적용 될 배치 문제의 성격을 분석하고, 3장에서 유전 알고리즘을 설계하고 적용한다. 4장에서는 시뮬레이션을 통하여 적용한 알고리즘의 성능을 평가하고 5장에서 결론을 맺는다.

2. 그레이팅 배치

그레이팅 설계 자동화 시스템에서 그레이팅은 정해진 배치 구획 안에 그레이팅의 폭과 남은 배치 구획의 폭을 고려하여 자동으로 분할 배치되는데 구획 내에 오픈이 존재 할 경우 오픈을 중심으로 분할을 하고 배치하게 된다.

오픈을 고려하여 분할 배치를 하게 되면 여러 제약 조건을 갖게 된다. 오픈의 중심을 지나며 배치 방향에 수직

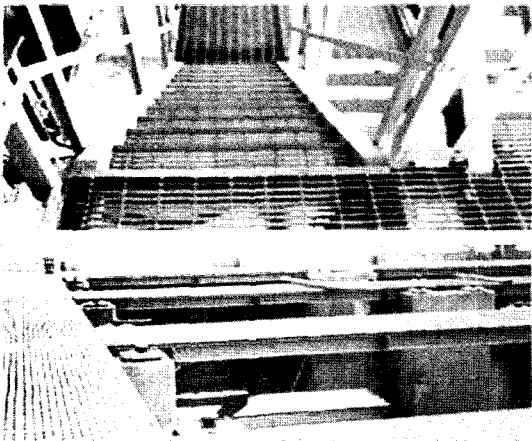


그림 1. 그레이팅

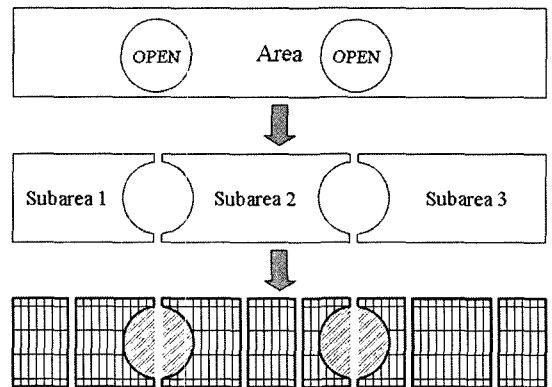


그림 2. 오픈이 존재하는 배치 구획의 분할

인 선을 c , c 에서부터 분할된 양쪽 그레이팅까지의 거리를 d_1 , d_2 , 그레이팅 사이의 거리를 g 라고 했을 때 배치를 만족하는 조건은 다음과 같다(그림 3).

이를 식으로 표현하면 다음과 같다. 알파벳 오름차순으로 우선순위가 높은 조건이다.

(a) $d_1 = d_2$, (1)

(b) $d_1 = 0 \vee d_2 = 0$, (2)

(c) $g = d_1 + d_2 \wedge d_1 \neq 0 \wedge d_2 \neq 0 \wedge d_1 \neq d_2$, (3)

(d) $(d_2 = d_1 + g \wedge 1 \leq d_1 \leq 13) \vee (d_1 = d_2 + g \wedge 1 \leq d_2 \leq 13)$ (4)

이러한 제약 조건 역시 제조 공정에서의 효율성과 설치시의 용이함으로부터 기인한다. 배치 구획상의 모든 분할은 위의 4가지 조건 중 하나를 만족해야 하며, 그림 3의 우선순위가 높은 조건을 많이 만족 하는 해가 최적 해이다.

3. 알고리즘의 설계 및 적용

유전 알고리즘은 1975년 John Holland가 제안한, 자연도태의 원리를 기초로 한 최적화(optimization) 방법으로 자연계에 있어서 생물의 유전(genetic)과 진화(evolution)의 메카니즘을 공학적으로 모델화하는 것에 의해 생물이 갖는 환경에서의 적응능력을 취급하는 것이다.^[6]

유전자(gene)란 유전정보를 담당하는 DNA 를 말한다. 특성의 유전자는 염색체의 특정 위치에 존재한다. 결국 유전정보는 염색체상에서의 위치와 염기의 배열에 의해 표현되는 것이다. 부모(parents)로부터 유전자에 의해 생물로서의 정보 전달이 행해지면 다음세대에는 각 개체 중에서

도 보다 우수한 즉, 환경에 적응도가 높은 개체의 유전 정보가 우선적으로 전해진다. 적응도가 낮은 개체는 수명이 짧고, 증식할 수 없게 되기 때문이다. 동시에 적응도가 낮은 종족도 자연 도태되어 간다. 이러한 원리에 기초하여 세대를 거듭해 가면 차례로 환경에 적응도가 높은 개체가 많아진다. 이것이 유전과 진화의 기본적인 원리이다^[7].

유전 알고리즘은 이런 유전과 진화원리를 이용하여 해가 될 수 있는 집단을 유지하고 재생산, 교배, 돌연변이와 같은 유전 연산자를 사용하여 진화시킨다. 유전 연산자는 최적의 해가 될 수 있는 집단을 선택하고 교배시키며 적응도를 평가함으로써 탐색을 실행한다^[8].

3.1 개체의 표현

분할 된 그레이팅의 폭은 가변적으로 정해지고 각각이 유기적으로 연결되어 전체의 폭을 이루게 되기 때문에 그레이팅 폭을 유전자로 설정하여 개체를 생성 할 경우 교배(crossover) 연산을 한 결과가 전체 폭과 일치하지 않는 문제가 생길 수 있어 적합하지 않다. 이를 해결하기 위해 오픈의 중심에 의하여 나뉜진 부분 구획을 하나의 유전자로 구분하고, 분할 배치 될 그레이팅의 폭을 원소로 갖게 하였다(그림 4). 이렇게 함으로써 각 개체들의 같은 순번의 유전자 길이를 고정시켜 교배 연산을 원활히 동작 시킬 수 있다.

k 세대(k 번째 반복횟수)에서의 집단(population) $P(k)$ 는 다음과 같이 표현된다.

$$P(k) = \{S_1(k), S_2(k), S_3(k) \dots, S_n(k)\} \quad (5)$$

유전자 A 는 w (분할 된 그레이팅의 폭)를 원소로 갖는 집합 형태로 결정되고, w 는 그레이팅의 사양과 규격을 고려하여 $400 \leq w \leq 1000$ 범위 내에서 임의의 정수 값으로 설정된다.

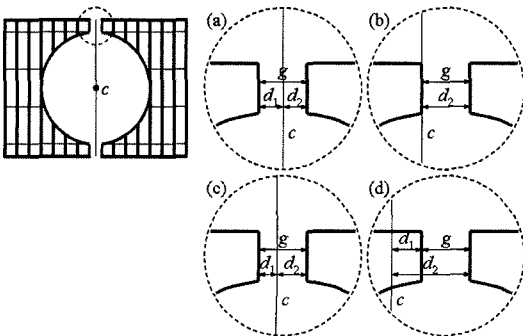


그림 3. 배치 구획의 분할 조건

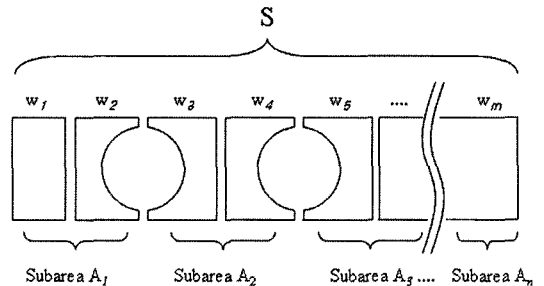


그림 4. Subarea를 유전자로 하는 개체의 표현

3.2 수행 연산

3.2.1 선택 (selection)

선택 연산은 다음 세대의 집단을 만들기 위하여 교배에 쓰일 두 개의 부모 해를 구하는 연산이다. 평가 함수를 통하여 나온 결과 값을 비교하여 선택 확률을 조정 할 수 있다. 선택 연산에는 일반적으로 많이 사용되는 룰렛 휠 선택(Roulette wheel selection) 기법을 사용하였다.^[8] 룰렛 휠 선택 알고리즘은 염색체의 적합도에 비례하여 개체를 선택하는 방법으로 적합도가 큰 개체가 선택될 확률이 높고 적합도가 낮은 개체는 선택될 확률이 상대적으로 낮게 된다.

3.2.2 교배 (crossover)

부모의 형질을 그 다음 세대에 물려주고, 더 나은 개체를 만들기 위해 교배 연산을 수행한다. 개체의 형태가 이진 스트링이나 실수형의 유전자가 아니므로 교배 연산 시에 구획 단위로 처리가 이루어지게 된다. 인접한 유전자간의 긴밀성이 성능에 영향을 미치므로 다점 교배(multi-point crossover)나 균등 분배(uniform crossover)등의 스키마 파괴가 잦은 기법보다는 단순 교배 연산(simple crossover)이나 이점 교배(two-point crossover)가 적합하다^[8]. 여기서서는 단순 교배 연산을 사용하였다.

3.2.3 돌연변이 (mutation)

일부 좋은 유전자형들이 열등한 개체에 있어 조기 소멸 되는 경우와, 비슷한 유전형질의 개체들로 인하여 때 이른 수렴이 일어 날 수 있는 경우를 방지하기 위하여 돌연변이가 연산을 수행한다.^[9] 여기서도 교배 연산의 경우와 마찬가지로 구획 단위에 돌연변이 연산이 적용되게 된다. 일정한 확률로 돌연변이가 결정되어진 유전자는 원소로 갖고 있던 w(분할 된 그레이팅의 폭)값을 새로 계산하여 재설정한다.

3.3.3 적합도 (fitness) 평가

세 연산을 거쳐 새로운 집단이 완성 될 때마다 개체들의 적합도가 평가된다. 평가 함수는 배치 구획의 분할 조건을 얼마나 만족하는지와 분할 배치 된 총 그레이팅의 수를 이용해 적합도를 계산한다. 그림 3에서 나타내었던 분할 조건 중 우선순위가 높은 조건을 많이 만족 할수록 좋은 해이기 때문에 각 분할 조건에 가중치를 설정하였고, 분할 된 그레이팅의 수가 적을수록(절단 횟수의 감소로 제조 공정상 비용 절감) 적합도가 높아지도록 설정하였다.

적합도 평가 함수는 다음과 같다.

$$f(x) = g(x) + h(x) \tag{6}$$

$$g(x) = \frac{(a*1) + (b*0.5) + (c*0.2) + (d*0.1)}{n} \tag{7}$$

$$h(x) = \frac{n}{m} \tag{8}$$

여기서 x 는 개체군의 세대수이고 a, b, c, d 는 분할 제약 조건, n 은 오픈의 개수, m 은 분할 배치 된 그레이팅의 수이다.

3.3 알고리즘 적용

위에서 설계된 알고리즘은 다음과 같이 배치 문제에 적용되었다. 우선 부분 구획을 유전자로 하여 표현한 개체들을 그림 5와 같이 단순 교배를 통하여 교배한다.

위 그림에서 개체 S_1 은 구획 $a_1, a_2, a_3... a_n$ 을 유전자로 하고 있고 그와 교배 연산을 할 S_2 는 구획 $b_1, b_2, b_3... b_n$ 을 유전자로 하고 있는 개체이다. 하나의 구획은 분할 조건에 의해 나뉘어진 다수의 그레이팅을 포함하고 있다. 현재 S_1 과 S_2 는 둘 다 분할 조건 3과 2를 앞에서부터 차례로 만족하는 형태를 띄우고 있다. 이렇게 같은 분할 오픈이 존재하더라도 여러 형태의 분할 그레이팅이 나타나며 이를 단순 교배 연산하여 나온 결과는 그림 6과 같다.

교배 연산을 통하여 새롭게 생성된 개체 S_{j1} 과 S_{j2} 는 각각 서로의 유전자를 하나씩 교환하였다. 그 결과 S_{j1} 은 분할 조건 1과 2를 만족하게 되고 S_{j2} 의 분할 조건 만족도는 변함이 없다. 상위의 분할 조건을 만족 할 수록 적합도가 높기 때문에 S_{j1} 은 교배 연산을 통하여 생존 가능성이 높아지게 되었다. 이렇게 생성된 새로운 개체는 적합도 평가 함수에 의해 평가를 받게 된다. 3.3.3절에서 선정한 평

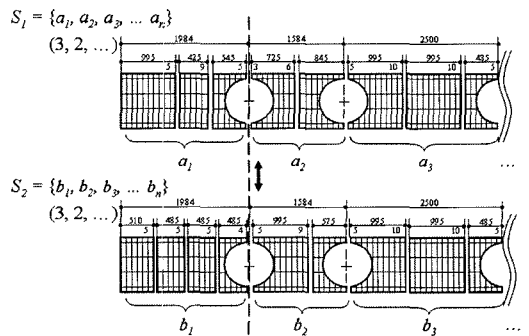


그림 5. 그레이팅 개체에 적용되는 단순 교배 연산

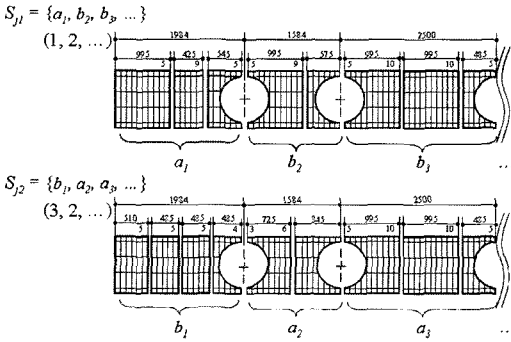


그림 6. 단순 교배 연산이 적용된 후의 개체

가 함수에 \$S_{j1}\$과 \$S_{j2}\$의 변수를 대입하면 다음과 같다

$$f(S_{j1}) = \frac{(1*1) + (1*0.5)}{2} + \frac{2}{8} = 1.00 \quad (9)$$

$$f(S_{j2}) = \frac{(1*0.5) + (1*0.2)}{2} + \frac{2}{9} = 0.57 \quad (10)$$

\$S_{j1}\$은 \$S_{j2}\$에 비해 적합도 평가 항목인 상위의 분할 조건을 만족하고 분할된 그레이팅의 수가 적기 때문에 적합도가 더 높게 나왔다. 이렇게 계산되어진 적합도에 따라 룰렛 휠 선택연산을 하여 다음 세대의 개체를 생성한다. 적합도가 종료 조건에 부합하면 그 개체는 최적해로 결정된다.

4. 성능 평가

시뮬레이션을 통하여 적용된 알고리즘의 효율성을 측정하였다. 한 세대를 30개의 개체로 구성하고, 총 400세대에 걸쳐 진화를 진행시켰다. 적합도에 따른 종료 조건은 주지 않고, 진화가 모두 끝나면 종료한다. 오픈의 개수를 20, 30, 40 개로 증가 시키면서 측정된 세대수에 따른 적합도 변화는 그림 7과 같다. 그래프를 보면 오픈의 개수가 증가함에 따라 수렴 속도가 늦어지게 되는데 20개 open의 경우 220구간에서, 30개 open의 경우 280구간에서 수렴 되는 것을 볼 수 있다.

기존까지의 그레이팅 분할 배치 문제는 숙련된 디자이너의 경험적 방법에 의존하여 해결해왔고, 이를 본 논문에서 제시한 알고리즘과 비교 평가 하였을 때, 결과는 그림 8과 같다.

실제 생산에 사용된 도면을 바탕으로 비교 평가를 할 때, 경험에만 의지하여 해결하는 방법은 오픈의 수가 많아질수록 효율적인 해결책을 찾지 못하고 비용이 증가

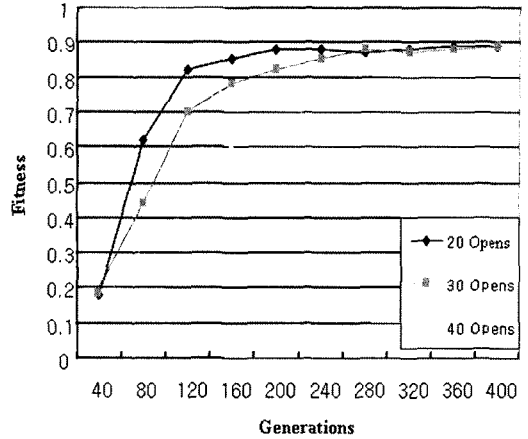


그림 7. 오픈의 수에 따른 적합도 변화

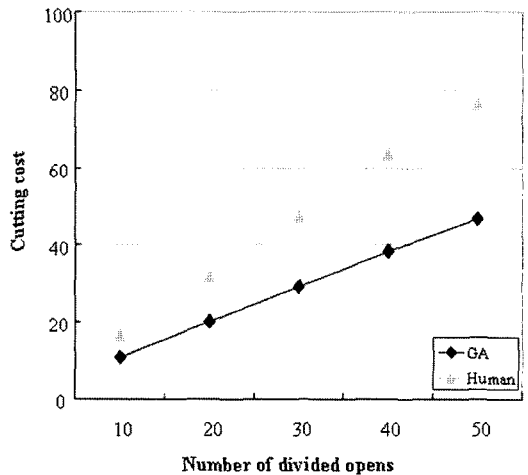


그림 8. 기존의 방법과 제시한 알고리즘의 비교

하였다. 반면 본 논문에서 제시한 방법은 비록 탐색 공간의 크기가 커짐에도 생산 비용을 절감하고 효율성을 만족하는 근사 해를 탐색해내었다

5. 결론

본 논문에서는 효율적인 그레이팅 배치 해를 찾기 위하여 유전 알고리즘을 적용하였다. 오픈으로 나누어지는 구역을 유전자로 표현하고, 선택, 교배 및 돌연변이에 의해 진화되며, 정의된 함수에 의해 유전자의 적합성을 평가한다. 수렴된 해는 분할 조건을 만족하는 근사해로서 상당히 효율적이었다.

유전 알고리즘의 장점은 크고 복잡하거나 제대로 이해

되지 않은 검색공간과 비선형 문제들에 대한 안정적인 검색 방법이라는 것이다. 성능 평가 결과 탐색 공간이 큰 경우에 대해서도 분할 조건을 만족하는 근사 해를 도출해냄으로써 상당히 효율적인 결과를 보여주었다. 그러나 진화에 있어서 초기해의 영향을 많이 받아 초기해가 근사 해에서 멀 경우 연산이 길어지거나 근사 해에 못 미치는 값을 수렴하는 경우가 생겼고, 개체군의 수가 적을 경우 국부해로 조기 수렴하는 경우가 발생 할 가능성이 있었다. 초기 해를 랜덤하게 결정짓는 것보다는 경험적으로 생성하도록 하여 성능을 더욱 개선시키는 방법에 대하여 연구를 계속 진행할 계획이다.

참 고 문 헌

1. Lee, K (1999), "Principles of CAD/CAM/CAE", Prentice Hall.
2. Li CL, Chan KW (1999), Tan ST. "A configuration space

approach to the automatic design of multiple-state mechanical devices." *Comput-Aided Des* 31, pp. 621-53.

3. Li CL, Li CG, Mok ACK. (2005), "Automatic layout design of plastic injection mould cooling system." *Comput-Aided Des* 37, pp. 645-62.
4. 송호정, 이범근 (2002), "최적 배치를 위한 유전자 알고리즘의 설계와 구현", *한국컴퓨터정보학회논문지*, Vol. 7, No. 3, pp. 42-48.
5. 장창두, 한운근 (2000), "개선된 유전자 알고리즘을 이용한 부재 절단경로 최적화에 관한 연구", *대한조선학회논문집*, Vol. 37, No. 3, pp. 90-98.
6. D.E. Goldberg (1989), "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley Publishing.
7. 문병로 (2003), "유전 알고리즘," *斗陽社*.
8. 진강규 (2000), "유전알고리즘과 그 응용", *교우사*.
9. Steve Ravin (2002), *AI Game Programming Wisdom*, Charles River Media.



이 정 규 (jglee@ece.skku.ac.kr)

2005년 강남대학교 전기전자컴퓨터공학부 학사
 2005년~현재 성균관대학교 정보통신공학부 컴퓨터공학과 석사과정

관심분야 : 모델링 및 시뮬레이션, 인공지능, ERP, USN



조 대 호 (taecho@ece.skku.ac.kr)

1983년 성균관대학교 전자공학과 학사
 1987년 Univ. of Alabama 전자공학과 석사
 1993년 Univ. of Arizona 전자 및 컴퓨터공학과 박사
 1993년~1995년 경남대학교 전자계산학과 전임강사
 1995년 성균관대학교 전기전자 및 컴퓨터공학부 조교수
 1999년 성균관대학교 전기전자 및 컴퓨터공학부 부교수
 2002년 성균관대학교 정보통신공학부 부교수
 2004년~현재 성균관대학교 정보통신공학부 교수

관심분야 : USN, 모델링 및 시뮬레이션, 지능 시스템, 네트워크 보안