

## SLS 쾌속조형장치를 위한 고속 패킹 알고리즘 개발

김부영, 김호찬(부산대학교 대학원 지능기계공학과),  
최홍태(경남정보대학 차량기계공학과), 이석희(부산대학교 기계공학부)

### A Rapid Packing Algorithm for SLS Rapid Prototyping System

Boo-Young. Kim, Ho-Chan . Kim(Department of Intelligence and Mechanical Engineering, Graduate school, Pusan National University), Hong-Tea, Choi(Vehicle Mechanics of Kyungnam College of Information and Technology), Suk-Hee, Lee(Department of Mechanical Engineering , Pusan National University)

#### ABSTRACT

With Rapid Prototyping system, the efficient packing in a fixed work volume reduces build time when mutiple parts are built in a process. In this paper, an efficient and rapid packing algorithm is developed for SLS system that has cylindrical workspace. A genetic algorithm is implemented to place as many part as possible in a vat. For fast computation, a collision detection algorithm "k-DOPs Tree" is implemented.

**Key Words :** Packing(패킹), SLS system(선택적 레이저 소결 장치), Genetic Algorithm(유전자 알고리즘), k-DOPs Tree(k- Discrete orientation polytopes tree),

#### 1. 서론

제품의 수명주기가 짧아지고 다품종 소량생산이 요구되어 신제품의 개발기간이 단축되어야 하는 추세에 따라 SLA(Stereo-Lithography Apparatus), SLS (Selective Laser Sintering), FDM(Fused Deposition Modeling), LOM (Laminated Object Manufacturing)등과 같은 여러 종류의 쾌속조형 장치가 쓰여왔다. 이러한 쾌속조형 장치들은 절삭가공이 단일 부품별로 가공을 진행하는 것과는 달리 여러 부품을 한번의 성형으로 제작할 수 있는 특징이 있다. 이 성형방법에서는 작업자가 작업공간에 부품의 형상정보를 효과적으로 배치함으로써 장비의 효율을 증대시킬 수 있다. 즉 작업자가 모델의 형상 정보를 성형공간 안에 빈공간을 최소화 하면서 중첩되지 않게 배치한 후 성형을 함으로써 장비 효율을 증대 시키는 장점을 가질 수 있다. 이는 곧 제품의 전체 가공시간을 단축시키는 효과를 가지며, 결국 비용 절감의 효과를 가지게 된다.

지지대가 필요한 공정을 가진 SLA, FDM 등은 지지대와 조형물이 z 방향으로 겹쳐서 표면 거칠기가 높아지므로 패킹보다는 배치가 중요했으나, SLS

나 LOM 등과 같이 지지대가 필요 없는 쾌속조형 기법은 장비별로 정해진 성형공간 안에 성형하고자 하는 부품을 최대한 많이 넣는 작업을 수행하므로 효율적인 패킹(packing)이 중요하다.

본 논문에서는 SLS 와 같은 원통형 작업공간을 가진 쾌속조형장치에서 성형시간 단축을 위해 3 차원 물체의 패킹을 빠른 시간 안에 수행하고자 하였으며, 특히 3 차원 형상을 나타냄에 있어서 BV-Tree (Bounding Volume Tree)를 사용하여 빠른 충돌 감지가 가능하도록 하였으며, 유전자 알고리즘을 통하여 최적의 해를 찾는다.

#### 2. 관련연구

Shian-Miin Hwang 등은 사각형의 배치문제에 유전자 알고리즘을 사용하였다<sup>[1]</sup>. 유전자를 나타내는 방식에는 두 파트와 그들의 배치관계를 나타내는 두 가지 연산자를 사용하고 파트와 연산자를 임의로 생성하고 올바르게 않은 유전자를 수정하는 알고리즘을 제시하였다.

C. Pimpawat 등은 3 차원 상자들을 층으로 나누어서 적재하고 각 층마다 적재 순서를 정함에 있어

서 협력공진화 유전자 알고리즘 (co-operative co-evolutionary genetic algorithm) 안의 특정 멤버들로 나타내었다<sup>[2]</sup>.

허성민 등은 복셀을 이용해서 3 차원 파트들을 SLS 에서 자동적인 조형자세 및 배치를 유전자 알고리즘을 통해서 결정하는 연구를 수행하였다<sup>[3]</sup>.

이러한 기존의 연구는 BL(Bottom-Left) 방식의 algorithm 등이 주로 사용되었으나 본 논문에서는 파트가 임의의 각도에서 접근하고 유전자 알고리즘을 이용해서 해를 탐색한다.

### 3. 3 차원 조형물의 고속 패키징

#### 3.1 고속 패키징 알고리즘

3 차원 패키징 알고리즘은 레이아웃과는 달리 vat 의 z 축으로 파트가 효율적으로 적재되어야 한다. 3 차원 패키징문제는 정확한 해를 찾을 수 없으나 조형 시스템의 오차 안에서 근사한 해를 찾을 수 있다.

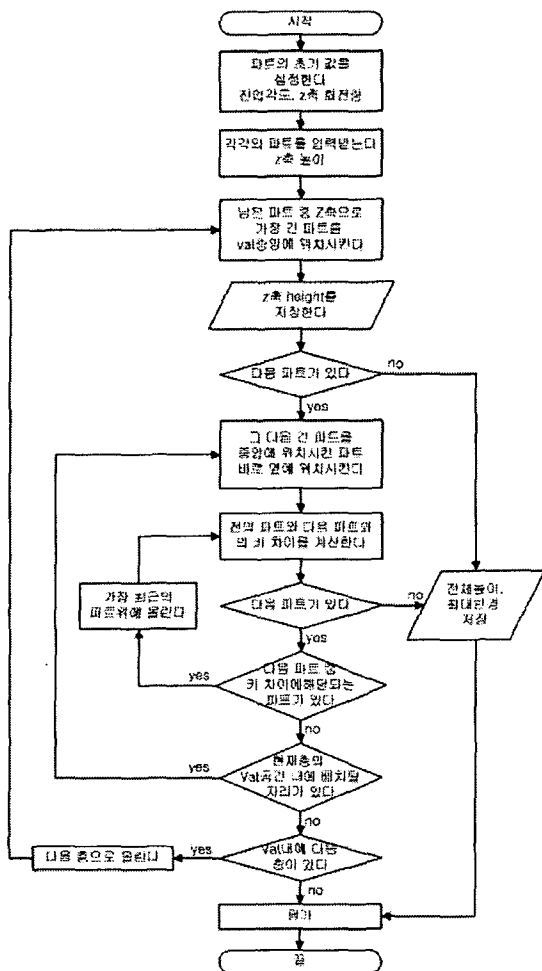


Fig 1. Rapid packing algorithm

본 알고리즘은 3 차원 패키징에 있어서 시스템에서 사용할 수 있는 해를 빠른 시간에 찾을 수 있는 Fig. 1 의 알고리즘을 제시한다.

이 고속 패키징 알고리즘은 각 파트의 초기값을 설정함으로써 시작된다. 성형방향은 각 파트에 알맞은 방식으로 정해져있으므로<sup>[4]</sup> 초기에 설정해

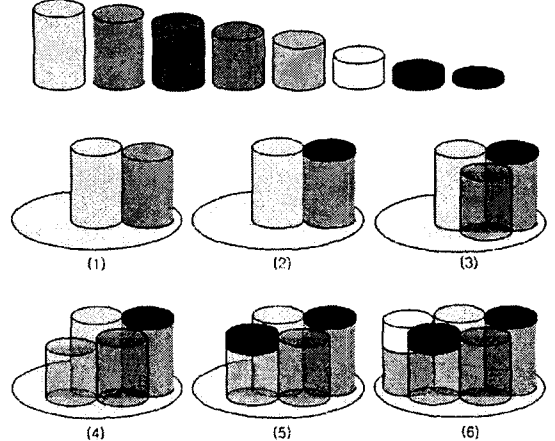


Fig 2. Example of packing order

야할 값은 파트가 원형 vat 에 진입하는 각도와 z 축의 회전량이다. 그 다음, 파트의 키를 입력받고 가장 키가 큰 파트를 vat 의 중앙에 위치시킨다. 이 위치시킨 파트의 키를 저장하고 남은 파트가 없을 때까지 중앙의 파트를 기준으로 다음 파트를 Fig. 2 의 (1)과 같이 임의로 정해진 각도로 충돌하기 전까지 접근시킨다. 이 접근한 파트와 가장 키가 큰 파트와의 키차이를 넘지않는 최대의 파트를 찾고 그러한 파트가 있으면 (2)와 같이 접근시킨 파트 위에 역시 임의의 각도로 접근하여 위치시킨다. 그러한 파트가 없으면 (3)과 같이 남은 파트중에서 가장 큰 파트를 다시 중앙의 파트에 접근시키고 다시 그 키차이에 해당하는 파트를 찾아 위에 올린다. 과정중에 남은 파트중 어떤 것도 vat 의 남은 부분에 들어갈 공간이 없으면 남은 파트중 가장 키가 큰 파트를 다시 다음 층의 중간에 위치시키고 앞서와 같은 방법으로 되풀이한다.

#### 3.2 k-DOPs Tree 를 이용한 Collision Detection

k-DOPs Tree 는 3 차원 형상을 k 개의 면을 갖는 Bounding Volume 을 써서 근사화해서 나타내는 방법이다<sup>[5]</sup>. Fig 3 는 Binary node 를 이용해서 나타낸 방법으로 처음 Head node 에서 분화된 중간 노드들과 말단의 Leaf node 까지 나누어져 있으며 Leaf node 는 face 정보가 저장되어 있다. 계층구조는 마지막 노드의 face 가 10 이하로 남을때까지로 설정했다.

두 node a 와 b 의 충돌을 감지하는 알고리즘은

Fig. 4 와 같다.

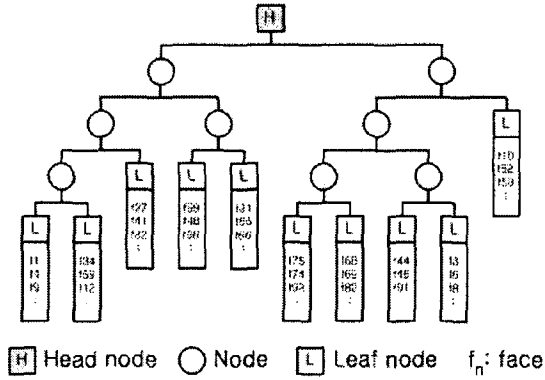


Fig 3. The structure of k-DOPs Tree

Simultaneous traversal of BV trees

a=k-DOPs of A's tree,  
 b=v of B's tree  
 a[i], b[j] children of a and b, resp

```

traverse(a, b):
a or b is not overlap → return
b is leaf →
a is leaf →
check primitives enclosed by a and b
return
    a is not leaf →
        for all i:
            a[i], b overlap → traverse(a[i], b)
    b is not leaf →
        a is leaf →
            for all i:
                a, b[i] overlap → traverse(a, b[i])
        b is not leaf →
            for all i:
                for all j:
                    a[i], b[j] overlap → traverse(a[i], b[j])
    
```

Fig. 4 k-DOPs tree Tranverse Algorithm

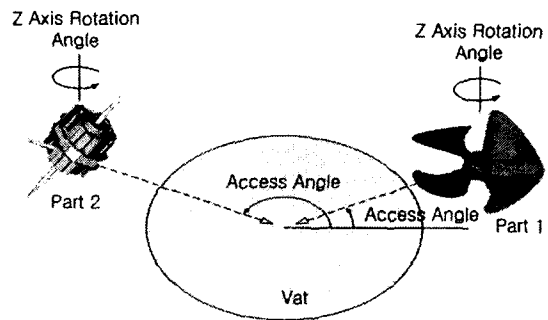
3.3 유전 알고리즘

유전자 알고리즘은 유전적 계승과 다윈적 생존 경쟁이라는 자연의 현상을 모델링한 해의 확률적인 탐색방법이다[9]. 본 연구에서는 빠르고 효율적인 해의 탐색을 위해서 가장 좋지 않은 유전자 10%와 가장 좋은 유전자 10%의 차이가 일정 이하가 될 때 해의 탐색을 종료한다.

3.3.1 유전자 구조

각 파트는 두 가지 gene 을 가진다. 하나는 vat 의 어느 각도에서 진입을 하게 되는 지를 나타내는 Access Angle 이고 다른 하나는 각각의 파트가 z 축 과 이루는 각도이다.

교배연산은 부모 유전자의 임의의 두 위치를 정 하고 그 사이에 있는 값을 서로 교배하여 자식 유 전자를 생성하며, 돌연변이연산은 임의의 위치에서 값을 변화시킨다.



Access Angle	23	154	11	44	95	174	...	7
Z Axis Rotation	12	105	38	84	74	54	...	122

Z 축의 키가 작은 순서

Fig 5. Chromosome encoding

3.3.2 목적 함수

packing 된 세대를 평가하는 평가함수는 vat 의 높이가 낮을수록 성형이 완성되는 시간이 적는데 기인하여 식 1 과 같이 packing 높이의 함수로 정한 다.

$$f(z) = \frac{H - z}{H} \quad (1)$$

여기서, H 는 vat 의 총 높이이고, z 는 packing 의 높이이다.

4. 적용 예

실험에서 설정한 RP 장비는 SLS system 인 Vanguard로 vat size 가 W370× D320× H445mm 이다. 실험을 위해서 파트의 크기를 적당히 스케일해서 적합한 모델들을 선정했다. 파트들은 STL 형식으로 입력 받았으며, 또한 가장 좋은 유전자 10%와 가장 나쁜 유전자 10%의 값이 평가함수의 5%이하로 수렴될 때 연산을 마친다. 다음 세대에 계승될 유전 자는 전체의 30%이며, 교배는 90%, 돌연변이는 5%

로 연산을 수행하였다.

실험에 사용된 파트는 Table. 1 과 같으며 위에서 아래로 키순서로 배열하였다. 수행된 결과는 Fig. 6 과 같다.







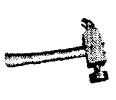

			
1: RP testimony 1	2: Motor cover	3: Pump housing	4: Fan locker
			
5: RP testimony 2	6: Fan blade	7: Hammer	8: Fan front cover

Table. 1 Sample parts

연산을 수행한 시스템은 PentiumIII-800Hz, 512M RAM 이며 연산시간은 파트에 따라 약간의 차이가 있으나 10 여초 내외이다.

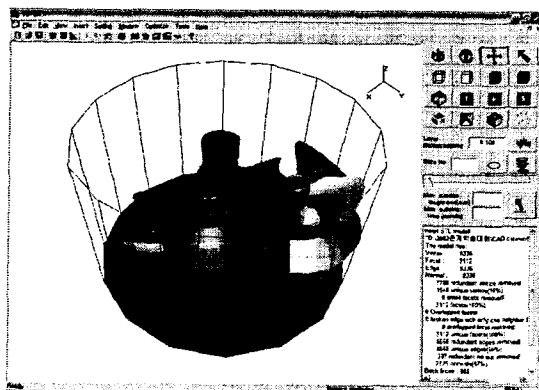


Fig.6 Result

## 5. 결론

본 연구에서는 SLS 를 위한 3 차원 형상의 고속의 패킹을 위한 알고리즘을 제시하였다. 또한, 정상상태 유전자 알고리즘과 유전자에 그룹을 만들어서 효과적인 최적화를 수행하였으며, 이에 필요한 유전자 구조와 평가함수를 제시하였다. 또한 각 배치의 빠른 평가를 위해 k-DOPs Tree 를 이용한 충돌 탐색을 실시하였다.

## 참고문헌

1. Shian-Miin Hwang, Cheng-Yen Kao, Jomg-Tzong

Horng, "On Solving Rectangle Bin Packing Problems Using Genetic Algorithms," Systems, Man, and Cybernetics, 1994. Humans, Information and Technology., 1994 IEEE International Conference on, Vol 2, pp.1583 -1590, 1994.

2. Pimpawat, C., Chaiyaratana, N., "Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem," Evolutionary Computation, Proceedings of the 2001 Congress on, Vol 2 , pp. 1197 -1204 vol. 2, 2001.
3. 허성민, 장복근, 최경현, 이석희, "SLS 에서의 자동적인 조형자세 및 배치 결정에 관한 연구," 한국정밀공학회지 제 16 권, 제 11 호, pp.139-147, 1999.
4. 허정훈, 이건우, "SLA 를 이용한 신속 시작작업에서의 최적 성형방향의 결정," 한국정밀공학회, 춘계학술대회논문집, pp.552-558, 1995.
5. Gabriel Zachmann, "Rapid Collision Detection by Dynamically Aligned DOP-Trees," Proc. of IEEE Virtual Reality Annual International Symposium; VRAIS '98. Atlanta, Georgia, 1998.
6. Zbigniew Michalewicz, "Genetic Algorithm + Data Structures = Evolution Programs," Springer-Verlag, 1996.