

SLC파일을 이용한 쾌속조형장치의 가공공정 개선에 관한 연구

김태호*, 김민주[†], 이승수[†], 이준희^{‡‡}, 전언찬^{***}

The Study on Manufacturing Process Improvement of Rapid Prototyping

by using SLC File

Tae Ho Kim*, Min Ju Kim[†], Sung Soo Lee[†], Jun Hee Lee^{‡‡}, Eon Chan Jeon^{***}

Abstract

This paper is described manufacturing to jewelry by using rapid prototype. At that time, the SLC file is useful for solved generating errors to part build file. Then, we have reduced the time of manufacturing complicated three-dimensional form. Generally, it is used the STL file for rapid prototype that it is necessary to considerable time when complicated form as jewelry generated part build file. But the SLC file is solved to problem because it consists of configuration of a section. Nevertheless, Generating Part build file have had a lot of problem when the SLC file is used.

Key Words : 쾌속조형장치(Rapid Prototype), STL 파일(STL file), SLC 파일(SLC file), 공정제작파일(Part build file)

1. 서 론

쾌속조형장치용 자료교환 표준으로 STL이라는 설계정보 교환 표준체계에 의거하여 운용되고 있다. 이는 설계된 제품형상의 기하학적 정보를 평면 삼각형들의 근사화된 집합으로 표현한 것이다. 그리고 STL 파일은 평면 삼각형들의 테이터만 포함하고 있기 때문에 간단하고 일반적인 삼각형 분할 알고리즘을 사용하여 정밀도를 쉽게 조절할 수 있다.⁽¹⁾⁽²⁾

하지만 STL 표준 파일 포맷은 여러 가지 단점을 가지고

있다. 데이터들의 중복으로 인한 데이터량의 증가와 굽곡진 곡면 등 곡면을 삼각형 평면들로 근사시켜 표현할 수 밖에 없는 한계, 위상정보를 갖고 있지 않기 때문에 부동 소수점연산으로 인한 수치 끝자리 오차 그리고 3차원 형상이 가지고 있는 유용한 정보를 STL파일로 변환하면서 기본적인 기하학적 형상정보만이 저장되어 오차의 원인이 되어 형상의 정밀도에 많은 영향을 미친다.⁽³⁾⁽⁴⁾⁽⁵⁾ 그리고 STL 파일의 용량은 적층 높이의 간격에 따라 같은 크기의 형상이라도 용량의 차이가 현격하게 나타난다.

일반적으로 STL 파일은 특정 프로그램에 의해서 자동으

* 발표자, 동아대학교 기계공학과 (kth0110@donga.ac.kr)
주소: 604-714 부산시 사하구 하단2동 840

+ 동아대 기계공학과

++ 동아대 신소재·화학공학부

*** 동아대 기계공학과

로 지지 구조물이 생성되어 쾌속조형장치에서 사용되어지는 파일로 변환되어 가공이 이루어진다. 이 과정에서 STL파일의 형상의 복잡성과 크기에 따라 지지 구조물 생성시간이 현격한 차이를 보이고 있다.

SLC파일은 3차원 모델을 높이에 따라 단면을 생성시킨 파일로서 STL파일이 가지고 있는 기본적인 형상정보를 가지고 있으므로 STL 파일의 용량의 10분의 1수준의 데이터 용량을 가지므로 아주 복잡하고 미세한 형상의 경우에 활용도가 높다. 그렇지만 테이터를 가공하는면에서 까다로워 많이 사용하지 않는다.

하지만 쥬얼리의 경우와 같이 복잡하고 작은 형상일 때 STL 파일의 용량으로 인하여 제작 시작단계에서 많은 시간이 소요된다.

그리하여 본 연구에서는 쥬얼리와 같은 복잡하고 미세한 형상에서 STL파일 대신 SLC 파일을 쾌속조형장치에서 사용하는 방법을 제안하고 복잡하고 미세한 형상에서의 가공시간을 단축하는 방법을 제시하고자 한다

2. 전체공정

쾌속조형체에서 3차원 형상제작의 전체공정을 살펴보면 3차원 모델링 파일을 일반적인 CAD 프로그램을 이용하거나 전문 변환프로그램을 이용하여 STL 파일로 변환한다. 그리고 부품준비 소프트웨어를 통하여 CAD모델의 검증, STL 파일의 오류보정 및 수정, STL파일의 추가 모델링, 지지구조물의 생성 및 적층과 리코팅 특성치를 설정하고 전체공정을 제어할 수 있는 파일로 변환하여 형상을 가공한다. 이 일련의 과정을 Fig. 1에 나타냈었다.

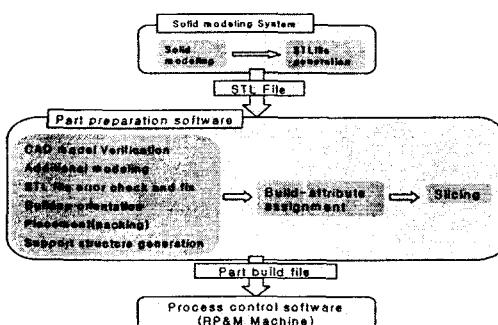


Fig. 1 Process of RP manufacturing

3. STL 파일과 SLC 파일

STL파일은 삼각형들이 연결되어 있는 메시로 물체를 표현하다. 이 삼각형들의 꼭지점은 어느 쪽 면이 볼륨의 내부에 속하는지 나타낼 수 있도록 일정한 방향으로 정렬되어 있다.

하지만 이런 꼭지점들이 하나 이상의 면에서 나타나기 때문에 꼭지점의 좌표값이 여러차례 기록되어 데이터의 양이 많아지는 단점을 가지고 있다.

하지만 SLC 파일은 STL 파일이 가지고 있는 단면의 좌표점만을 가지고 있기 때문에 STL 파일에서 가지는 문제점을 일부 해결하고 데이터의 양에서도 상당히 줄일 수 있는 장점을 가지고 있다.

4. 실험방법

운영시스템이 없이 컴퓨터가 작동을 할 수 없는 것과 같이 모든 쾌속조형장치에서도 운영소프트웨어가 필요하다. 이 과정에서 슬라이스 높이에 따라 STL파일의 데이터 양이 기하급수적으로 늘어난다. 이를 쾌속조형장치에서 사용할 수 있는 파일로 전환하는 작업에서 상당히 많은 시간을 보내야만 한다. 그래서 SLC 파일을 이용하여 Part build file을 생성하고 STL파일을 이용한 Part build file 생성시 걸리는 시간과 비교하고 이렇게 만들어진 두 가지 Part build file을 이용, 적층조형하여 형상을 비교해보고자 한다. 일련의 과정을 도식화하여 Fig. 2에 나타내었다.

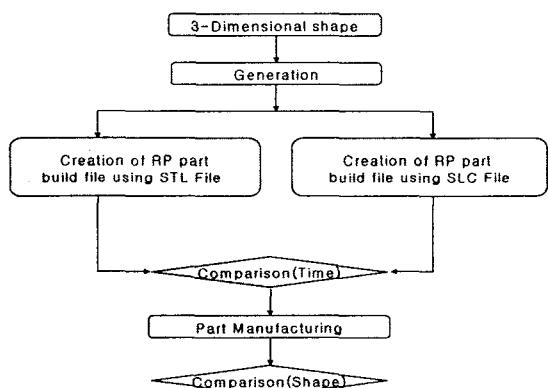


Fig. 2 Flowchart of experiment

4. Part build file 제작

4.1 STL 파일

STL 파일은 채속조형장치의 기본 파일포맷으로 사용되어 많은 연구 개발이 이루어져 왔고 많은 소프트웨어들이 존재 한다. 그 중에서도 3D system사의 STL 파일을 이용한 Part build file제작 과정을 살펴보면 STL 파일을 불러 사용자가 편하게 볼 수 있도록 디스플레이를 해준다. 그리고 자동으로 지지 구조물을 생성해주고 이렇게 생성된 자동구조물의 크기 및 적층높이 리코팅 특정치의 설정 등을 수행하여 Part build file을 생성시킨다 이와 같은 과정을 도식화하여 Fig. 3과 같다.

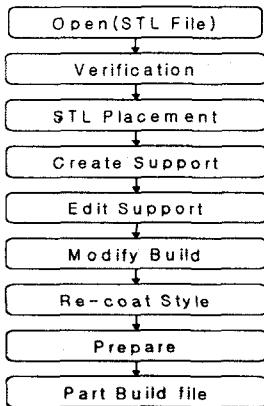


Fig. 3 Transformation process of part buile file using STL File

4.2 SLC 파일

SLC 파일은 STL파일을 적층높이에 따라 단면으로 슬라이싱한 것으로 삼각형 면 균사화로 인해 생기는 방대한 데이터의 양을 줄일 수 있었다. 그리고 Part build file로 변환하는 과정을 살펴보면 다음과 같다. 먼저 3차원 형상 설계시 형상의 변형이 생기지 않도록 지지구조물을 생성시키고 SLC 파일을 변환할수 있는 프로그램에서 SLC 파일로 변환을 한다. 그리고 SLC 파일을 지지해주는 지지구조물을 생성하기 위하여 베이스를 STL 파일로 생성시키고 플래폼과 구조물을 재거하기 위한 지지구조물을 만들어준다. 그 다음과정은 STL파일을 생성하는 방법과 동일하게 만들어 Part build file을 생

성할 수 있다. 이와 같은 과정을 Fig. 4에 나타내었다.

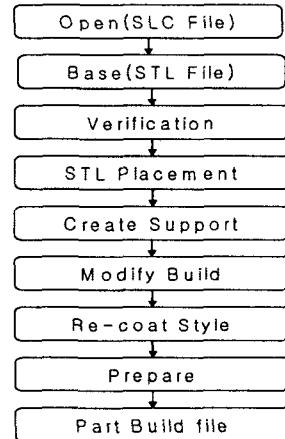


Fig. 4 Transformation process of part buile file using SLC File

4. Part build file 생성

Fig. 5는 실험에 쓰인 쥬얼리 데이터로 무작위로 추출하여 여섯가지의 모델을 크기와 형상을 그림으로 나타내었다. 주로 글자를 음각으로 형성하였다.

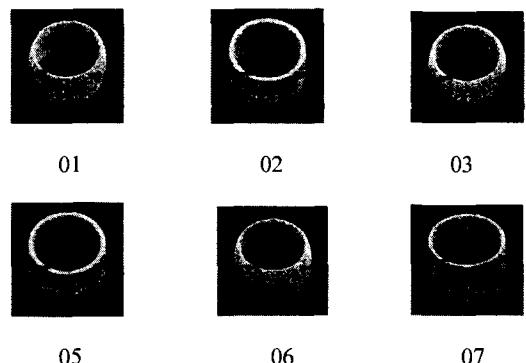


Fig. 5 Data of experiment (STL File)

Table 1은 데이터의 크기를 비교하기 위하여 작고 미세한 형상을 가지고 있는 쥬얼리 모델링을 가지고 파일크기를 비교하였다. 최대 5.4배에서 3.4배 정도차이가 나고 평균적으로 4배의 차이가 났다.

Table 1 Comparison of File size (STL File, SLC File)

샘플	STL파일 크기	SLC 파일크기
01	5,120 KB	1,124 KB
02	5,190 KB	1,299 KB
03	5,590 KB	1,601 KB
04	5,040 KB	1,289 KB
05	4,358 KB	1,270 KB
06	6,133 KB	1,128 KB

Fig. 6은 샘플번호에 따른 Part build file 생성시 걸리는 시간을 나타낸 것으로 파일의 크기보다는 파일의 적층 높이가 높을수록 더 오래 걸리는 것을 볼 수 있었다. STL파일을 이용한 경우 평균적으로 4시간 이상이 걸렸으며 SLC 파일을 이용한 경우 평균적으로 1시간 미만의 시간이 걸려 작업 효율 면에서 좋은 결과를 나타내었다.

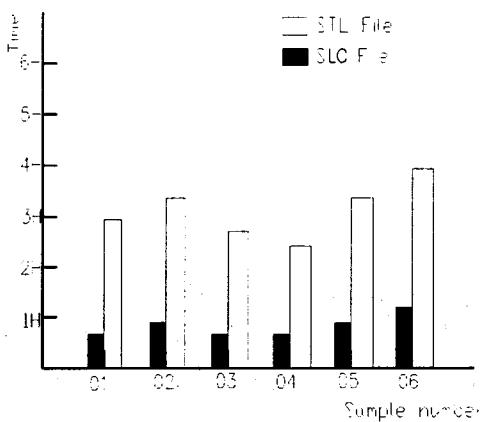


Fig. 6 sampling rate(for sample)

Fig. 7은 샘플의 개수에 따라 Part build file을 생성하는데 걸리는 시간을 나타낸 그래프이다. SLC 파일을 사용한 경우 점진적으로 증가하는 모양이나 STL 파일을 사용한 경우 4개 이상의 경우 급속하게 증가하는 것을 볼 수 있었다. 이는 컴퓨터의 부동소수점 연산으로 인해 단면생성시간이 개수의 증가로 전체 공정에서 걸리는 시간을 증가시킴으로써 전체공정의 시간증가로 보여진다.

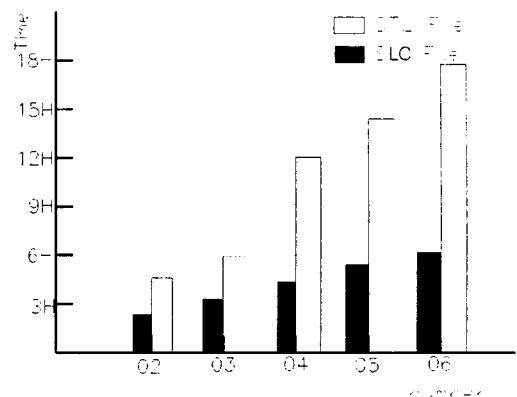


Fig. 7 Sampling rate(for number)

Fig. 8은 b)는 SLC 파일을 이용하여 얻어진 결과물로 a)의 STL파일과 비교해서 차이를 보이지 않았으며 전체공정에서 시간을 단축시킬 수 있었다.



Fig. 8 Comparision of Shape (STL File, SLC File)

5. 결론

쥬얼리와 같은 특정형상의 경우 STL 파일을 이용하여 쾌속조형장치에서 제작하는 경우와 SLC 파일을 이용하여 제작하는 두 가지의 경우를 서로 비교·분석하여 다음과 같은 결론을 얻었다.

1. Part build file 생성 시 STL 파일이 가지는 시간상의 문제점을 파악하였다.
 2. 복잡한 형상의 경우 STL 파일대신 SLC 파일을 제안하였으며 SLC 파일을 이용할 경우 상당한 시간 단축을 가져올 수 있었다.
 3. 여러개의 반지를 성형하기 위한 Part build file을 제작하기 위해서는 STL 파일보다는 SLC 파일을 이용하여 제작하는 것이 효율적으로 시간을 활용할 수 있었다.

참 고 문 헌

- (1) H. C. Chae, 1997, "Creation of Topological Information from STL Using Triangle Based Geometric Modeling", KSPE, Vol 14, No. 2, pp 136-144
- (2) H. T. Choi, S. H. Lee, 1996, "A Study on Error Verification of STL Format for Rapid Prototyping System ", KSPE, Vol 13, No. 10, pp 46-55
- (3) J. B. Park, Y. J. Son, S. K. Kim, E. C. Jeon, J. W. Kim, 1999 " A Study on DB Construction for Error Modification of STL Format and Efficiency by Shape Restoration", KSPE, Vol.16, No.9, PP. 143 ~ 148
- (4) T. S. Myung, H. C. Chae, O. H. Kim, 1996 " Application of Delaunay Triangulation on RP" KSPE, Vol. 8, No. 3, PP. 35 ~ 41
- (5) M. S. Hong, K. H. Shin, S. G. Son, 2000, "A study on RP Part Production Using Intersecting Circle Approximation Method" Vol. 9, No. 1, PP. 68 ~ 74