

## 근사된 원호를 따르는 CL-카테산 가공

주 상 윤  
산업공학과

### <요 약>

CL-카테산 방식은 산업현장에서 자주 사용되는 가공 경로방식이다. 본 논문은 CL-카테산 가공에서 원호보간법으로 CL 데이터를 얻는 전반적 방법을 다루고 있다. 곡면과 절단 평면간의 교선으로부터 CL점들을 찾는 절차와 얻어진 CL점으로부터 원호를 근사시키는 방법이 제시되었다. 본 방법은 허용오차 내에서 최소의 원호들로 CL점들을 근사시키고 있으며 근사된 원호들의 갯수는 CL데이터 간의 간격에 의하여 거의 영향을 받지 않는다.

---

## CL-Cartesian Machining along Approximated Circular Arcs

Ju, Sang Yoon  
Dept of Industrial Engineering

### <Abstract>

A method of CL-Cartesian machining is often applied in industry for machining path planning. This thesis deals with an overall procedure for obtaining CL data by circular interpolation in the CL-Cartesian machining. An algorithm is introduced for finding CL data on intersection curves between a parametric surface and a cutting plane. And a method is studied for determination of circular arcs approximating CL data. The method generates circular arcs of minimal number within given tolerances. The number of circular arcs is hardly influenced by intervals between two adjacent CL points.

## 1. 서 론

곡면을 가공하기 위한 경로계획은 곡면의 등경곡선을 따라 가공이 이루어지는 파라메트릭 방식과 가공곡면과 절단평면의 교선을 따라 가공이 이루어지는 카테산 방식으로 크게 나뉘어진다. 파라메트릭 방식은 카테산방식에 비하여 수치적 계산은 간단하지만 가공할 곡면이 위상학적으로 불규칙한 혼합곡면(Compound Surface)인 경우 사용하기 곤란하므로 작업현장에서는 대부분 카테산 방식에 의한 가공경로가 사용되고 있다. 또한 카테산 가공 방식은 공구접촉점(Cutter Contact Point : CC점)이 절단평면 상에 존재하는 CC-카테산 방식과 공구위치점(Cutter Location Point: CL점)이 절단평면 상에 존재하는 CL-카테산 방식으로 분류된다. CC-카테산방식은 가공곡면과 절단평면의 교선으로부터 CL점을 얻는 방식으로 개념적으로 이해하기는 쉬우나 가공 중에 공구가 좌우로 흔들려 쉽게 부러지는 경향이 있다[1]. 반면에 CL-카테산방식은 가공곡면을 공구의 반지름만큼 윽셋하여 절단평면으로 자른 교선으로부터 CL점을 얻는 방식이다. 이 방식은 CC-카테산 방식에 비하여 계산이 다소 복잡하고 개념적 이해가 까다롭지만 공구가 항상 절단평면을 따라 안정되게 움직이므로 작업자들에 의하여 선호되고 있다.

본 연구에서는 파라메트릭 함수식으로 정의된 자유곡면에 대하여 원호보간을 이용하여 CL-카테산 방식의 가공경로를 찾는 전반적인 과정을 다루고 있다. 일반적으로 NC기계에서 공구의 이동은 직선보간(G00,G01)과 원호보간(G02,G03)의 기능을 사용한다. 이 때 직선보간보다는 원호보간을 이용하는 것이 NC테이프의 길이도 줄이고 보간오차도 줄일 수 있는 것으로 알려져 있다[2]. 원호보간은 공구의 CL데이터가 XY-평면, YZ-평면, XZ-평면의 주평면에 평행하게 존재해야 가능하므로 2차원 NC가공에서 주로 이용된다. 반면에 자유곡면의 밀링가공과 같은 3차원 가공에서는 가공 경로를 파라메트릭 방식으로 취하거나 CC-카테산 방식을 취하는 경우 CL점들은 일반적으로 평면 상에 존재하지 않으므로 직선보간을 사용하게 된다. 하지만 가공 경로가 CL-카테산 방식을 따르는 경우 절단평면을 XZ-평면이나 YZ-평면에 평행하게 고정시키면 얻어진 CL점들이 절단 평면 상에 존재하게 되므로 원호보간을 사용할 수 있다.

## 2. 원호근사에 의한 CL-카테산 방식의 가공경로

3축 NC밀링에서 플랜드밀 공구를 사용하여 곡면을 가공하는 경우 CC점과 CL점의 관계는 그림 1과 같고 다음의 식으로 나타낼 수 있다.

$$r_L = r_c + R(n - u) \quad (1)$$

- 단,  $r_L$  : CL점
- $r_c$  : CC점
- $R$  : 공구의 반지름
- $n$  : 곡면의 단위 법선벡터
- $u = (0,0,1)$  : 공구의 회전축 단위벡터

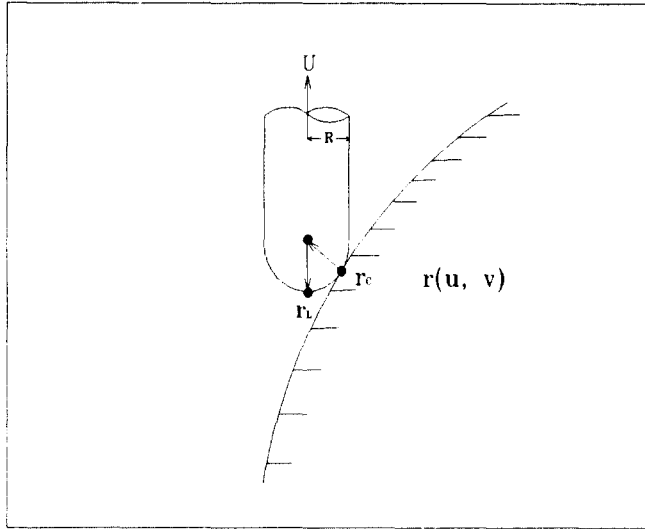


그림 1. CC점과 CL점

CL-카테산 가공에서 CL점과 공구의 중심점은 항상 절단평면 상에 존재하게 된다. 따라서 원호로 CL점들을 근사하는 절차는 다음의 세 단계로 이루어진다.

- 1) 윗셋곡면과 절단평면의 교선을 구한다.  
가공이 이루어지는 동안 공구의 중심점들은 가공곡면을 공구의 반지름만큼 윗셋시킨 곡면 상에 존재하므로 윗셋곡면과 절단평면과의 교선을 찾는다. 얻어진 교선은 공구 중심점의 가공경로에 해당된다.
- 2) 교선상의 점들로부터 CL점들을 구한다.  
교선 상의 점들은 공구의 중심점이며 교점들을 -Z축 방향으로 공구반지름만큼 이동시키면 CL점들을 얻게된다.
- 3) CL점들을 다수의 원호들로 근사시킨다.  
허용오차의 범위 내에서 가능한 많은 CL점들을 근사시키는 원호를 찾는다.

### 3. 윗셋곡면과 절단평면의 교선

가공할 곡면이 매개변수 함수식  $r(u,v)$ 로 정의되었다고 하자. 이 때 곡면  $r(u,v)$ 를 공구의 반지름  $R$ 만큼 법선벡터 방향으로 윗셋시킨 곡면  $r^o(u,v)$ 은 다음과 같다.

$$r^o(u,v) = r(u,v) + Rn(u,v) \tag{2}$$

단,  $n(u,v)$  : 곡면  $r(u,v)$ 에 대한 단위 법선벡터

편의상 CL-카테산가공을 위한 절단평면이 XZ-평면에 평행하게 설정되었다고 하자. 윗곡면  $r^0(u,v)$ 와 절단평면의 교선은 다음의 두 단계로 결정된다.

### 1) 초기교점의 결정

초기교점은 윗곡면과 절단평면 간의 교선을 찾기 위한 출발점이다. 윗곡면의 경계선들과 절단평면의 교점들이 초기교점에 해당된다.

### 2) 교선의 추적

초기교점으로부터 출발하여 곡면 상에 존재하는 모든 교선들을 찾아낸다.

## 3.1 초기교점의 결정

윗곡면의 경계선들을 다수의 선분들로 이루어진 다각형으로 근사시킨 다음 절단평면과의 교점을 찾는다. 이렇게 얻어진 교점들은 근사점에 불과하므로 수치적 방법으로 윗곡면 경계선과 절단평면 간의 정확한 교점을 찾는다. 정확한 교점은 교선의 양끝점으로서 교선의 추적을 위한 출발점으로 이용되며 초기교점목록(Initial Intersection Point List)에 저장된다.

## 3.2 교선의 추적

초기교점으로부터 출발하여 교선 상의 점들을 연속적으로 찾아가는 과정이다. 그림 2와 같이 초기교점에서 윗곡면  $r^0(u,v)$ 와 절단평면에 대한 각각의 단위법선벡터를  $m$ ,  $n$ 이라 할 때 교선 방향의 단위 추적벡터  $v$ 는 다음과 같다[9].

$$v = (m \times n) / |m \times n| \quad (3)$$

단,  $m$  : 윗곡면  $r^0(u,v)$ 의 단위법선벡터

$n$  : 절단 평면의 단위법선벡터

식(3)에서 동일한 파라미터값  $(u,v)$ 에 대하여 윗곡면  $r^0(u,v)$ 의 단위법선벡터와 원곡면  $r(u,v)$ 의 단위법선벡터는 항상 일치한다[3]. 또한 절단평면은 XZ평면으로 가정하였으므로 단위법선벡터  $n$ 은 Y축 방향의 단위벡터  $(0, 1, 0)$ 이다.

그림 2와 같이 출발점에서 결정된 교선의 추적벡터  $v$ 와 적절한 크기의 스텝길이(step length)  $\delta$ 로부터 예상점  $G$ 를 구한 다음, 점  $G$ 에 대하여 최단거리에 위치한 교선 상의 점을 Jacobian inversion에 의하여 찾는다[9]. 얻어진 교점을 다시 출발점으로 하여 동일한 과정을 반복하면 교선 상에 있는 점들을 연속적으로 얻을 수 있다. 이에 관한 보다 자세한 내용은 Barnhill[4]과 Ju[5]를 참조하라.

추적과정에서 얻어진 교점이 윗곡면의 경계선을 벗어난 경우 교선에 대한 추적은 중단된다. 교선은 다수의 교점들로 정의되므로 일단 발견된 교선은 교점들의 linked list 구조로 교선목록(Intersection Curve List)에 저장된다. 교선이 발견될 때마다 교선의 양끝점들을 초기교점목록에서 제외시킨 다음, 남은 초기교점들 중에서 임의의 점을 취하여 다른 교선을 추적한다. 초기교점목록에 교점들이 더 이상 존재하지 않을 때까지 교선의 추적을 반복하면 지정된 절단평면과 윗곡면 간에 존재하는 교선들을 모두 찾을 수 있다.

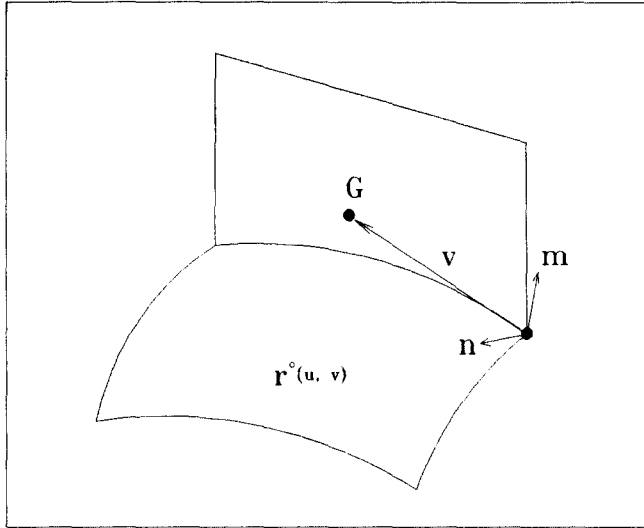


그림 2. 교선의 추적을 위한 예상점 G의 결정

### 4. CL점들의 원호 근사

교선목록에 저장된 교점들은 공구의 중심점이므로  $-Z$ 축 방향으로 공구의 반지름만큼 이동시키면 CL점들을 얻을 수 있다. 연속적으로 얻은 CL 점  $p_1, p_2, \dots, p_n$ 들을 하나의 원호로 근사시킨다고 하자. 그림 3과 같이 근사원호의 양끝점을 점  $p_1$ 과  $p_n$ 으로 정하면 원호와 나머지 CL점  $p_i (i = 2, \dots, n-1)$ 들간의 오차  $\epsilon_i$ 들은 다음 식으로 정해진다.

$$\epsilon_i = |p_i - O| - \rho \tag{4}$$

단,  $O = M - \rho \cos(\theta)u$  : 원호의 중심점

$\rho = L/\sin(\theta)$  : 원호의 반지름

$M = (p_1 + p_n)/2$

$L = |p_n - p_1|/2$

$a = (a_1, a_2, a_3) = (p_n - p_1)/|p_n - p_1|$

$u$  :  $a$ 에 수직인 단위벡터 중 점  $M$ 에서 내부CL점  $p_2, p_3, \dots, p_{n-1}$ 을 향한 벡터

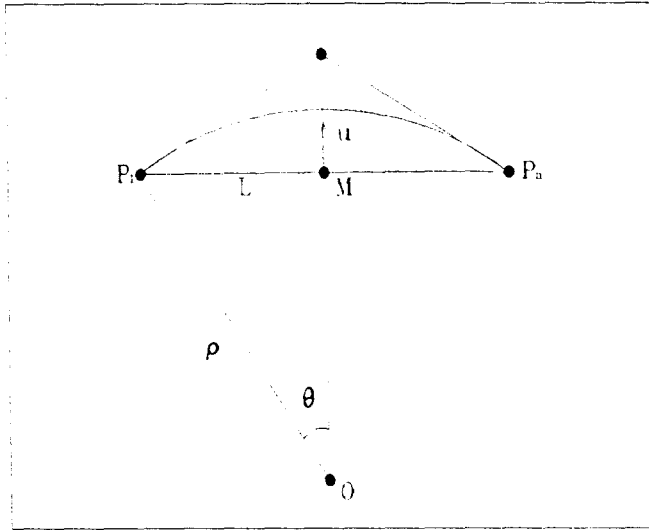


그림 3. CL점들에 대한 근사원호

원호의 반지름  $\rho$ 에 대하여 CL점  $p_1, p_n$ 을 양끝점으로 가지는 원호는 항상 두개가 존재한다. 이들 두원호 가운데 내부 CL점  $p_2, p_3, \dots, p_{n-1}$ 에 가까운 원호를 택한다. 설정된 원호와 CL점들간의 최대오차를 원호오차, 그리고 최대오차가 존재하는 CL점과 원호의 중심점  $O$ 와의 거리를 최대오차점거리라고 부르기로 하자. 우선 원호의 반지름  $\rho$ 의 초기값을 임의로 정한 다음 최대오차점거리가 원호반지름보다 큰 경우에는 원호의 반지름을 증가시키고, CL점거리가 원호반지름보다 작은 경우에는 원호의 반지름을 감소시킨다. 이러한 과정으로부터 최소의 원호오차를 가진 원호반지름  $\rho$ 는 쉽게 찾을 수 있다.

연속적으로 얻어진 CL점들을 다수의 연결된 원호들에 의하여 근사시키는 방법을 살펴보기로 한다. 우선 하나의 원호를 근사시키기 위한 CL점들을 CL점구름이라고 부르기로 하자. 만약 CL점구름에 속한 CL점들이 근사된 원호에 대하여 허용오차를 벗어나지 않는다면 CL점을 하나 더 추가하여 다시 새로운 근사원호를 구한다. 반면에 CL점 구름에 속한 점들 가운데 어느 하나라도 근사원호에 대하여 허용오차를 벗어나는 경우, 마지막으로 추가된 CL점을 제외시킨 직전 원호를 근사원호로 정한다. CL점들로부터 근사원호를 구하는 과정은 다음의 알고리즘을 따라 이루어진다.

단계 1. CL점구름에 존재하는 점의 갯수가 3개 이하이면 CL점들을 추가한다.

추가할 CL점이 존재하지 않으면 알고리즘을 중단한다.

단계 2. CL점구름에 있는 CL 점들을 원호로 근사시킨다.

단계 3. 만약 원호의 오차가 허용오차를 벗어나면

직전원호를 근사원호로 취하고 CL점 구름에 마지막 2개의 CL점만을 남기고 단계 1로 간다.

만약 원호의 오차가 허용오차를 벗어나지 않으면 단계 4로 간다.

단계 4. 만약 CL점구름에 CL점을 더 추가할 수 없으면 알고리즘을 중단하고 그렇지 않으면 다음 CL점을 CL점구름에 추가하고 단계 2로 간다.

그림 4는 위의 알고리즘을 이용하여 CL점들을 원호로 근사시킨 예를 보여준다. CL점들은  $\times$ 와  $\square$ 로 표시되었으며 이들 가운데 특히  $\square$  표시는 근사된 원호의 양끝점을 가리킨다. 그림 4에서 CL점들은 42개 주어졌으며 7개의 원호들에 의하여 근사되었다.

그림 5는 자유곡면을 CL-카테산 방식으로 가공한 경로를 보여준다. 그림 5(a)의 자유곡면은  $5 \times 5$  퍼거슨 합성곡면(Composite Surface)으로 모델링되었으며 그림 5(b)는 자유곡면과 다수의 XZ-절단평면에 의하여 구한 CL점들을 원호들로 근사시킨 가공경로를 도시하였다.

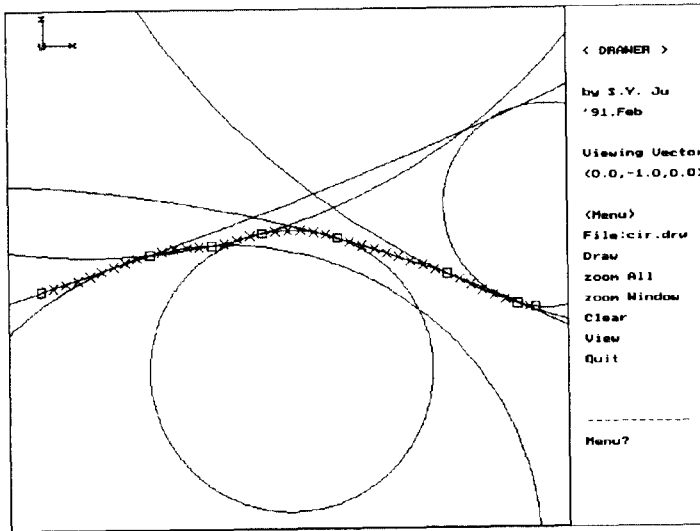
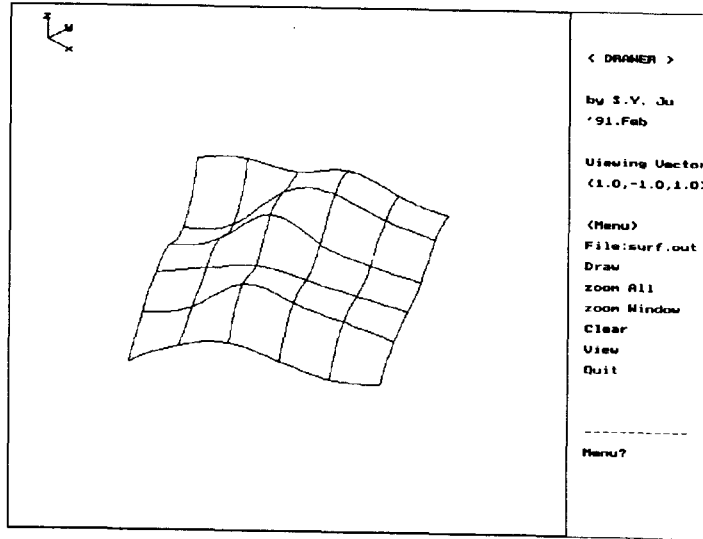
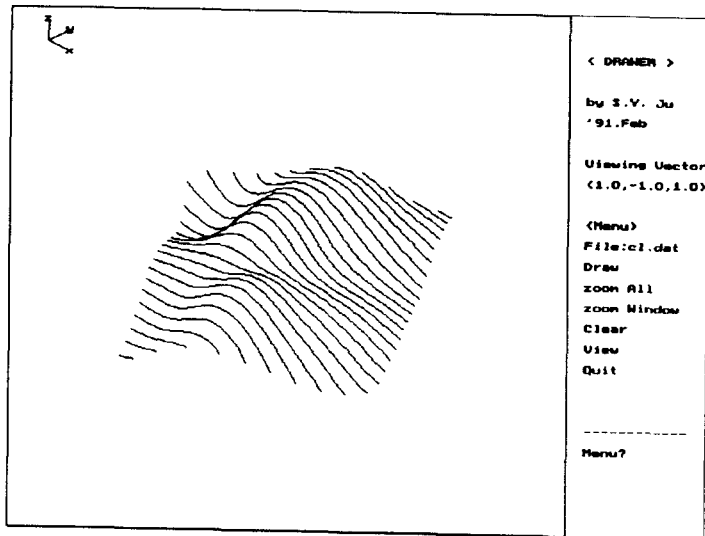


그림 4. CL점들을 근사한 원과 원호



(a)



(b)

그림 5. 퍼거슨 합성곡면(a)과 원호근사에 의한 CL-카테산 가공경로(b)



### 5. Biarc모델과 원호근사의 비교

biarc모델은 평면 상에 존재하는 이웃한 두점들 사이를 부드럽게 연결된 두개의 원호로 보간하는 기법으로 2차원 평면 상에 존재하는 점들을 원호로 보간하기 위하여 자주 이용되는 방법이다[6]. biarc 모델은 NC가공에 있어서 원호보간을 이용할 수 있으므로 점들을 수학적 스플라인으로 보간하는 방법보다 효율적인 것으로 알려져 있다[2]. biarc 모델이 2차원 XY-평면 상에 주어진 점들을 보간하는 데에는 효과적이지만 절단평면과 곡면 간의 교선으로부터 구한 CL점들을 보간하는 데에는 문제점이 있다. 이미 앞에서 언급했듯이 곡면과 절단평면간의 교선은 다수의 교점들로 정의되며 스텝길이에 따라 교점들간의 간격과 교점들의 수가 정해진다. 직선보간에서는 경제적인 NC데이터를 얻기 위하여 최적의 스텝길이를 정하는 방법이 연구되어 있으나 biarc모델의 경우에는 그와 같은 연구가 이루어진 바 없다[7]. 스텝길이를 크게 정할 경우 얻어진 교점들은 교선을 제대로 반영하지 못할 우려가 있으므로 현장작업자들은 작은 스텝길이를 이용하여 교점간의 간격을 짧게 하는 경향이 있다. 이런 경우 보간할 교점은 많아지게 되며 biarc모델에 의하여 보간된 원호의 수도 비례하여 증가한다.

이에 반하여 본 방법은 허용오차를 기준하여 근사원호를 생성하므로 점들의 간격에 의하여 근사된 원호의 수가 거의 영향을 받지 않는다는 장점이 있다. 참고로 그림 5(b)에서 생성된 다수의 CL점 경로 가운데 하나를 택하여 스텝길이와 허용오차의 변화에 따라 근사된 원호의 갯수들을 조사한 결과는 표 1과 같다. 스텝길이가 짧아짐에 따라 biarc모델에서 원호의 갯수는 증가하는 데 반하여 본 방법으로 근사한 원호의 수는 거의 변화가 없었다. 또한 본방법은 허용오차가 변함에 따라 원호의 수가 증가한다. 하지만 biarc모델의 원호의 수와 비교해보면 여전히 훨씬 적은 갯수의 원호를 생성하고 있다.

표 1. 스텝길이와 허용오차에 따른 근사원호의 수

스텝길이	CL점의 수	허용오차에 따른 본 방법의 원호 수				biarc모델의 원호 수
		$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$	
1.0	14	6	6	6	6	26
0.5	26	7	12	12	12	50
0.2	62	7	14	26	29	122
0.1	123	7	14	32	56	244
0.05	244	7	14	29	71	486
0.02	607	7	14	29	67	1212
0.01	1213	7	14	29	68	2424

## 6. 결 론

본 논문에서는 파라메트릭 함수식으로 정의된 자유곡면에 대하여 원호보간을 이용하여 CL-카테산 방식의 가공경로를 찾는 전반적인 과정을 제시하였다. 곡면과 절단평면간의 교선으로부터 CL카테산 방식의 가공을 위한 CL집들을 찾는 절차가 소개되었고 얻어진 CL 점으로부터 원호를 근사시키는 방법이 제시되었다. 본 방법은 허용오차를 만족시키는 가운데 최소의 갯수를 가진 근사원호를 생성하고 있다. 또한 얻어진 근사원호는 CL집들간의 데이터 간격에 거의 영향을 받지 않으므로 CL-카테산 가공에서 유용하게 이용될 수 있다. 본 연구는 이미 개발되었거나 개발 중인 CAM 시스템에 적용할 수 있을 것으로 기대된다.

## 참 고 문 헌

1. Hwang, J.S., "Interference-free tool-path generation in the NC machining of parametric compound surfaces", Vol.24, No.12, 1992, pp.662-676.
2. 최 병규, CAM시스템과 CNC절삭가공, 청문각, 1989.
3. Farouki, R.T., "The Approximation of Non-Degenerate Offset Surfaces," CAGD, Vol.3, 1986, pp.15-43.
4. Barnhill, R.E., Farin, F., Jordan, M., and Piper, B.R., "Surface/surface intersection", CAGD, Vol.4, No.1, 1987, pp.3-16.
5. Ju, S.Y., Construction of Blend Surfaces in Surface Modeling, KAIST, PhD Dissertation, 1989
6. Bolton, K.M., "Biarc Curves", CAD, Vol 7, No 2, 1975, pp89-92.
7. 김대현, 최병규, "자유곡면 절삭을 위한 경제적인 CL 데이터 계산", 대한산업공학회지, Vol.9, No.2, 1983, pp.27-35.
8. 김 창효, 단면곡선 분할을 이용한 Faring에 관한 연구, KAIST, 석사학위논문, 1991
9. Choi, B.K., Surface Modeling for CAD/CAM, Elsevier, 1989.