

4축 밀링가공을 위한 간섭없는 공구경로 생성에 관한 연구

서 석 환

· 이 기 상

포항공과대학교 산업공학과

Interference-Free Tool Path Generation for Four-Axis NC Machining

Suk-Hwan Suh

Kee-Sang Lee

Department of Industrial Engineering

POSTECH, Pohang-Korea, 790-600

요 약

4축 자유곡면가공에 있어서 공구간섭 문제는 가공형상의 취득성 뿐만 아니라 가공의 정밀도를 위하여 매우 중요한 고려사항이다. 공구의 접근자세가 고정된 3축 밀링가공의 공구간섭에 관한 기존의 많은 연구가 있지만 이러한 연구는 공구의 접근자세가 변하는 4축 밀링가공에의 적용이 곤란하다. 왜냐하면 곡면의 형상에 따라 공구몸체의 곡면사이의 간섭문제가 부각되며, 이에 따른 공구자세의 창출문제가 해결되어야 하기 때문이다.

따라서 본 논문에서는 4축가공에서 발생할 수 있는 공구간섭을 유형별로 분류, 각 간섭들을 발견하고 해결할 수 있는 방법의 전개와 이 방법을 이용하여 공구경로를 산출하고 그래픽 시뮬레이션과 실제의 시험절차를 통하여 제시된 방법의 유효성을 검증한다.

1. 서 론

설계된 자유곡면의 형상을 주어진 허용오차 안에서 밀링머시인으로 정밀하게 가공하기 위해서는 공구의 운동을 정의하는 일련의 공구경로를 생성할때, 공작기계의 형태에 따라 곡면과 공구사이의 간섭을 고려하는 것이 중요하다.

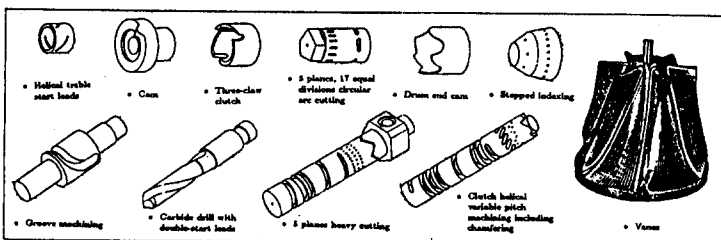


그림 1. 회전형상의 부품들

그림1과 같은 회전형상의 자유곡면을 가지는 부품들은 가공시에 공작물의 회전운동과 직교좌표상에서 공구의 곡선보간운동을 필요로한다. 이러한 운동조건을 만족하고 공작물의 한번 셋업으로 곡면 전체를 가공할 수 있는 밀링머시인은 그림2와 같이 공작물을 회전시키는 회전축과 공구의 곡선보간운동을 가능케하는 직교3축들로 구성된다. 참고로 이러한 4축 머시인은 기존에 있는 3축 머시인에 회전축을 부가 장착하여 경제적으로 쉽게 구성할 수 있다. 이전에 우리는 회전형상의 자유곡면을 설계할 수 있는 기하모델러, 공구경로 산출 모듈, 산출된 공구경로의 검증을 위한 그래픽 시뮬레이터, 그리고 포스트프로세서(Post Processor)로 구성된 4축CAM시스템을 개발하였다[1]. 그러나 위에서 제시한 공구경로 산출 알고리즘은 공구간섭을 고려하지 않았으므로 제한적인 부품의 형상에 대해서만 간섭없는 공구경로산출이 가능하다. 따라서 본 논문에서는 회전형상의 자유곡면가공을 위한 4축 밀링 머시인의 간섭없는 공구경로산출 알고리즘을 개발한다.

공구간섭이 없는 공구경로를 산출하기 위해서는 곡면의 형상과 공작기계의 형태에 따라 적당한 공구의 크기, 공구경로상에서의 진행간격, 공구의 접근자세 및 공구위치가 결정되어야 한다. (주지할 것은 4축 밀링 머시인일 경우에 공구의 접근자세는 공구에 대한 공작물의 회전각으로서 이는 공구에 대하여 상대적으로 공작물이 고정되어 있고 공구가

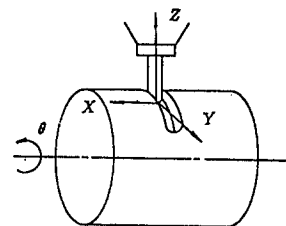


그림 2. 4축 밀링 머시인

공작물 주위를 자유롭게 회전하여 임의의 자세를 유지한다고 볼 수 있기 때문이다.)

공구간섭에 관한 기존의 연구들을 살펴보면 Hansen과 Arbab[2]은 곡면을 삼각형들로 나누어 근사시킨 후 부분분할(subdivision)방법을 사용하여 공구가 임의의 위치와 고정된 자세로 곡면을 접근하여 곡면에 가장 먼저 접촉하기 위한 공구의 위치를 찾는 알고리즘을 제시하였다. 이방법은 공구자세가 이미 고정된 3축 밀링머시인에 쉽게 적용되어 하나의 공구접촉점에 대한 공구의 위치를 전체곡면에 대하여 간섭이 없게 결정할 수 있다. Lee, et al.[3]은 곡면의 특정부위에 대하여 공구의 크기 때문에 볼엔드밀(ball end mill)의 볼(ball)부분에서 생기는 간섭(2절 참조)을 체크하는 알고리즘을 제시하였고 Choi, et.al[4]은 곡면을 다각형으로 근사시킨 후 CC(cutter contact)경로를 따라 곡면의 형상에 대하여 적당치 못한 공구의 진행에 의한 간섭과 공구의 크기에 의한 간섭(TSI)을 발견하고 제거하는 방법을 보였다. Chen과 Ravani[5]는 곡면에 대해 최소자승법을 이용하여 오프셋 곡면을 구하고 교차 투프가 없는 상태에서 공구경로를 산출하여 TSI를 제거한다. 이방법은 TSI를 피할 수는 있지만 많은 계산시간과 구해진 오프셋 곡면이 부정확하다는 단점이 있다.

이와같이 기존에 연구된 이론은 공구자세가 고정되어 공구자세를 결정할 필요가 없는 3축 밀링가공인 경우에는 유용하나 공구자세를 결정해주어야하는 4축 밀링가공에서 발생하는 모든 간섭들을 해결하지는 못한다. 왜냐하면 공작물이 회전함에 따라 상대적으로 공구의 자세가 변하는 경우에 곡면에 대한 공구의 접근자세가 적당하지 않으면 간섭(TAI: tool axis interference)이 생기기 때문이다. 그러므로 4축밀링머시인에서는 TAI가 없게 공구의 자세(공작물의 회전각)를 적당하게 결정하는 것은 매우 중요하다. 따라서 본 논문에서는 다른 간섭들과 함께 TAI를 집중적으로 다룬다.

이 논문은 다음과 같은 순서로 전개된다. 2절에서는 4축 밀링가공시 발생할 수 있는 3가지 유형의 간섭들을 원인별로 분류하고 3절에서는 제시된 간섭유형들을 해결하는 알고리즘을 제시한다. 4절에서는 제시된 알고리즘을 컴퓨터에 구현하여 부품의 곡면에 대하여 생성된 공구경로의 그래픽 시뮬레이션과 실제의 시험절삭을 통하여 연구한 알고리즘의 유효성을 검증한다. 마지막으로 본 논문의 5절에서는 결론 및 요약으로 끝을 맺는다.

2. 4축밀링가공에서의 간섭

일반적으로 볼엔드밀이 주어진 CC점((S(u,w))을 접근하기 위한 공구끝점은 아래와 같다.

$$P_t(u,w) = [x_t, y_t, z_t]^T \\ = S(u,w) + R[N(u,w) - A].$$

여기서 R은 공구반경, N(u,w)는 S(u,w)에서의 단위법선벡터를 나타낸다. 3축밀링머시인의 공구자세 A는 항상 [0,0,1]^T로 고정되어 있으므로 공구끝점이 CL데이터가 된다. 따라서 3축인 경우 곡면의 기하학적 형상의 특성에 따라 간섭이 없게 적당한 공구의 크기, 경로상에서의 공구진행간격, 그리고 공구위치를 결정해주면 된다. 그러나 4축밀링머시인에서는 CC점에 대한 공구자세 A를 먼저 결정하고 공구 끝점을 구해야한다. 그러므로 4축에서는 3축과 달리 CL데이터를 구할때

공구의 간섭이 없게 공구자세도 결정해야한다. 4축인 경우에 A는 CC점에 대해서 유일하게 결정되지 않으므로 A를 임의의 초기자세로 유지시킨후 공구의 자세에 의한 간섭이 없도록 조정해야한다. 여기서 우리는 앞으로 알고리즘의 제시를 위해 초기자세 A를 X축과 수직이고 CC점에 대한 공구중심점 C[xc, yc, zc]를 향하는 단위법선벡터로 정의한다.

$$A = \frac{1}{\sqrt{y_c^2 + z_c^2}} [0, y_c, z_c]^T, \quad (1)$$

여기서 공구중심점 C = S(u,w) + R·N(u,w)이다. 실제로 4축 밀링머시인은 공구의 자세가 Z축에 고정되고 공작물이 회전하므로 CL데이터는 다음과 같이 공구의 끝점과 공작물의 회전각을 포함한다.

$$CL(u,w) = [P_t(u,w), \theta(u,w)]^T = [x_t, y_t, z_t, \theta]^T$$

위의 회전각 θ는 상대적으로 공작물과 공구가 자세 A를 유지하기 위하여 공작물이 회전하는 각이다.

$$\theta = \tan^{-1}(y_c/z_c).$$

공작물이 회전한후 공구가 CC점을 접근하기 위한 공구끝점의 위치는 다음과 같다.

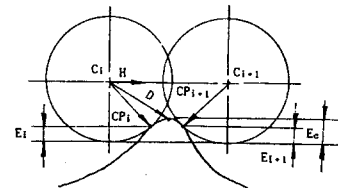
$$P_t = (S(u,w) + R[N(u,w) - A]) \text{Rot}[X, \theta].$$

위에서 보인 방법대로 CL데이터를 구할 때 4축밀링머시인에서 발생하는 간섭은 그 원인에 따라 a)공구진행간격에 의한 간섭(SSI), b)공구크기에 의한 간섭(TSI), c)공구자세에 의한 간섭(TAI)으로 나눌 수 있다.

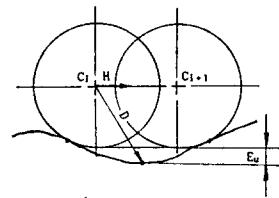
2.1 진행간격에 의한 간섭

곡면상의 곡선을 따라 그림3(a)와 같이 불룩한 부분에서 공구가 현재의 CC점인 P_i에서 다음 CC점인 P_{i+1}으로 진행할 때 간섭이 생기고 그림3(b)와 같이 오목한 부분에서는 미절삭부분이 생긴다.

그림3(a)에서 C_i, C_{i+1}는 공구접촉점 P_i = S(u_i, w_j), P_{i+1} = S(u_{i+1}, w_j)에 대한 공구중심점이고 P_c = S(u_c, w_j)는



a. convex



b. concave

그림 3. 진행간격에 의한 간섭

P_i 와 P_{i+1} 사이의 중간점으로서 $u_c = (u_i + u_{i+1})/2$ 이다. 그리고 공구의 진행방향을 나타내는 단위벡터 H 는

$$H = \frac{C_{i+1} - C_i}{|C_{i+1} - C_i|}$$

이고 간섭량 E_o 는 아래와 같다.

$$E_o = \max\{E_i, E_{i+1}, E_c\}, \quad (2)$$

여기서

$$\begin{aligned} E_i &= R(1 - \sqrt{1 - (N_i \cdot H)^2}), \\ E_{i+1} &= R(1 - \sqrt{1 - (N_{i+1} \cdot H)^2}), \\ E_c &= R - \sqrt{D \cdot D - (H \cdot D)^2}. \end{aligned}$$

위와 비슷하게 그림3(b)에서 미절삭량 E_u 은

$$E_u = \sqrt{D \cdot D - (H \cdot D)^2} - R \quad (3)$$

이다.

만약 위에서 제시한 간섭량이 허용오차보다 클 때 SSI가 발생한다고 볼 수 있고 이 때 진행각을 반으로 줄여 허용오차보다 작아질 때까지 반복계산하여 확인한다. 다음은 SSI를 발견하고 제거하는 과정을 보여준다.

Porcedure 1: SSI

1. Read in P_i and P_{i+1} .
2. Compute N_i and H .
3. if $N_i \cdot H < 0, N_{i+1} \cdot H > 0$, then compute $E = E_o$ by Eq.(2). Otherwise compute $E = E_u$ by Eq.(3)
4. if $E \leq \epsilon$, Stop. Otherwise $P_{i+1} = P_c$, goto 2.

2.2 공구크기에 의한 간섭

이 간섭은 그림4와 같이 곡면의 오목한 부분을 반경이 큰 공구로 접근할 때 생기는 간섭으로 공구의 반경을 줄이면 제거할 수 있다. 간섭이 일어나는 공구의 부위에 따라 공구의 끝부분에서 발생하는 볼부위간섭(ITB: intersection of tool ball)과 공구의 몸체에서 발생하는 몸체간섭(ITC: intersection of tool cylinder)으로 분류된다.

ITB는 CC점의 곡률에 의해 발견할 수 있다. 만약 CC점의 곡률이 $1/R$ 보다 크면 공구의 끝부분에서 간섭이 일어난다. 곡면의 접촉부위에서의 곡률 $K(u, w)$ 는 다음과 같다.

$$K(u, w) = \frac{[u', w']^T D [u', w']}{[u', w']^T G [u', w']^T}$$

여기서 $u' = du/dt$, $w' = dw/dt$ 이고 G , D 는 곡면의 첫번째, 두번째 기초행렬(fundamental matrix)이다.

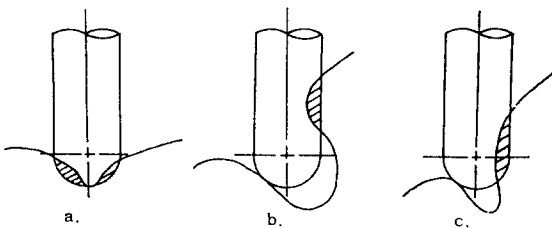


그림 4. 공구크기에 의한 간섭

그림4(b)-(c)에 보여진 ITC는 곡률로 발견할 수 없다. 이러한 간섭은 CC경로를 따라 교차루프가 있는지를 확인하여 발견한다. 이 발견과정은 3절에서 자세히 다룬다.

2.3 공구자세에 의한 간섭

이 간섭은 그림5와 같이 곡면위의 접촉점에 대해 공구의 접근자세가 잘못된 경우를 보여주고 있다. 그림5(b)의 경우는 자세가 변하지 않고 공구의 반경을 줄임으로 이러한 간섭을 제거할 수 있다. 그러나 이러한 경우는 공구의 크기를 줄이는 것보다 자세를 변화시키는 것이 절삭효율의 측면에서 훨씬 낫다. 공구자세에 의한 간섭을 발견한다는 측면에서는 [2]에서 제안한 방법을 이용할 수 있다. (CC점을 접촉하기 위한 공구중심점과 주어진 자세로 공구가 곡면을 향해 이동할 때 곡면에 제일 처음 접촉하기위한 공구중심점을 [2]의 방법을 이용하여 계산하고 비교한다.) 그러나 공구자세를 변화시켜 TAI를 제거한다는 측면에서는 기존에 연구된 이론을 적용할 수 없다는 것과 공구자세를 어느 정도 변화시켜야 간섭을 제거할 수 있는 가라는 문제가 생긴다. 이 문제는 3절의 알고리즘에서 자세히 다루어진다.

3 공구간섭의 해결방법

이 절에서는 주어진 일련의 CC점들의 경로상에서 발생하는 공구크기에 의한 TSI와 공구자세에 의한 TAI를 해결하는 방법을 보인다. 여기서 제시하는 방법은 2절의 SSI 알고리즘에 의하여 공구의 진행각은 적절하다는 가정하에 전개된다.

3.1 TSI의 해결방법

TSI는 CC평면상에서 공구경로의 교차루프를 확인하여 발견할 수 있다. u, w 영역에서 정의된 곡면위에서 CC경로가 $w = w_i$ 에서 u 를 따르면 CC평면은 다음과 같이 정의된다. CC경로상에 존재하는 어떤 한 점 $P = S(u_0, w_i)$ (P 는 현재공구의 접촉위치)를 지나고 YZ평면에 평행한 평면을 CC평면으로 한다. CC경로상에 있는 점들에 대한 공구중심점경로를 CC평면에 투사시키면 2D 곡선을 얻게되고 이 곡선상에서 P_p (P 를 CC평면에 투사시켜 얻은 점)를 중심으로 교차루프의 존재를 확인하여 TSI를 발견한다.

위에서 제시한 방법은 개념적으로 단순하고 ITB와 ITC에도 적용할수있다. 그러나 곡면의 형상이 복잡할 수록 원래의 오프셋곡선은 CC평면에 투사된 오프셋곡선과 차이가 심해져 위 방법의 간섭 발견 능력은 줄어든다. 그러므로 이 방법

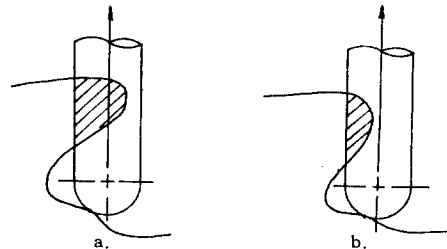


그림 5. 공구자세에 의한 간섭

의 간섭발견능력은 a) CC경로가 CC평면으로부터 벗어나 있는 정도, b) CC경로상에있는 CC점의 단위법선벡터가 CC평면과 벗어나 있는 정도에 의존한다.

w가 X축방향과 일치하는 회전형상의 곡면에 대해서는 첫 번째와 같은 문제는 발생하지 않는다. 그러나 곡면의 형상이 복잡하여 두번째 문제의 정도가 심하면 공구가 CC경로가 아닌 방향으로 간섭을 일으킬 수 있다.

위에서 논의한 것을 기초로 TSI의 프로시듀어를 나타내면 다음과 같다.

Procedure 2: TSI

1. Define CC-path as $S(u,w)$, and CC-plane as $x=P_x$, where P_x is x component of $S(u_0,w_j)$.
2. Form an offset curve composed of m segments: $s_i = C_i \cdot C_{i+1}'$, $i=1, \dots, m$, where C_i' is the tool-center point at $S(u_i,w_j)$ projected onto the CC-plane.
3. For $i = 1, \dots, m$
 - a. If $K(u_i,w_j) > 1/R$ (where the curvature is for the curve given by $S(u_i,w_j)$), then remove $S(u_i,w_j)$ from the CC-path.
 - b. If s_i intersect with s_{i+1}, \dots, m , remove $S(u_i,w_j)$ from the CC-path.

3.2 TAI의 해결방법

공구자세에 의한 간섭을 제거하기 위해서는 각각의 CC점에 대해서 공구의 접근이 가능한 자세를 선택해야한다. 접근가능한 자세를 얻기 위해서는 먼저 CC점에 대해서 접근가능한 자세의 영역을 구하고 그 영역안에 존재하는 한 자세를 선정한다.

3.2.1 접근가능한 자세

접근가능한 공구자세영역이란 곡면의 한 점을 공구가 접근할 때 곡면과 공구사이엔 간섭없이 공구가 취할 수 있는 모든 접근 자세의 집합을 말한다. 접근가능한 자세를 구하기 위해서는 공구의 접촉점에 대한 공구중심점 C를 구하고 이 중심점을 지나고 YZ평면과 평행한 CL평면을 설정한 후 그림6과 같이 C를 중심축으로 하여 공구가 자유롭게 CL평면상에서 움직일 수 있는 영역을 구한다. 이영역은 C를 중심으로 T_1 과 T_2 벡터로 이루어지는 cone이다. 여기서 T_1 과 T_2 는 공구가 CL평면과 곡면사이의 Intersection Curve(IC)와 교차하지 않으면서 움직일 수 있는 영역의 경계벡터를 나타낸다. 그리고 이 때의 IC는 곡면과 평면사이의 IC를 구하는 알고리즘을 이용하여 구한다.[6] 여기서 한가지 주지할 것은 접근가능영역의 의미가 곧 이 영역안에 존재하는 모든 자세가 간섭없이 접근 가능하다는 뜻이 아니고 그 영역의 일부분만이 접근가능하다는 것을 나타낸다. 이 영역의 일부분만이 접근가능한 이유는 즉, 그림6(b)와 같이 CL평면과 곡면이 교차하는 부분에서의 곡면단위법선벡터가 이 CL평면상에 존재하지 않는 경우가있기 때문이다. 따라서 곡면의 형상이 복잡할 수록 구해진 접근가능영역의 크기보다 실제로 접근가능한

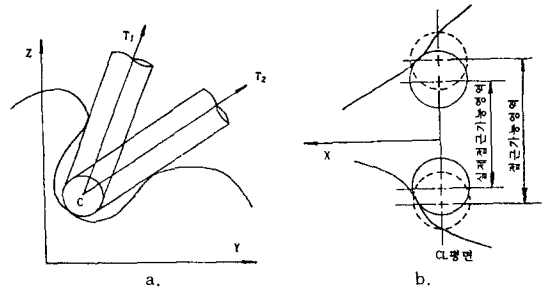


그림 6. 접근가능한 자세영역

영역이 작아지므로 접근가능성이 큰 자세를 선택하기 위해 각 CC점에 대한 공구자세는 다음과 같이 구해진 T_1 과 T_2 의 중간자세를 취한다.

$$A = (T_1 \cdot |T_2| + T_2 \cdot |T_1|) / (|T_1 \cdot |T_2| + T_2 \cdot |T_1|) \quad (4)$$

여기서 공구자세를 위와 같이 중간값을 취해도 곡면의 형상에 따라 접근가능하지 못할 때가 있다. 이러한 문제를 해결하기 위해서는 [2]에서 제시한 방법을 이용하여 구해진 공구자세에 대한 공구위치를 구한다.

위에서 논의한 것을 기초로 접근가능한 공구자세를 구하는 과정은 다음과 같다.

1. 공구접촉점 P에대한 공구중심점 C계산.
2. C의 x성분을 기초로 CL평면 설정
3. CL평면과 곡면사이의 IC 계산(혹은 Curve상에 존재하는 점 Q_i 계산, $i = [1:n]$).
4. 접근가능한 자세영역의 경계인 T_1, T_2 계산.
5. 접근가능한 자세 A결정.

위의 과정 중에서 1,2,3,5는 쉽게 구할 수 있으므로 여기서는 T_1, T_2 를 구하는 과정을 자세히 다룬다.

3.2.2 T_1 과 T_2 를 구하는 방법

여기서 제시하는 알고리즘은 IC상에 존재하는 점 Q_i , $i \in [1:n]$ 와 그점에 대한 곡면의 단위법선벡터 N_i 가 구해졌다는 가정하에서 전개된다. 접근가능한 자세영역의 경계인 T_1, T_2 를 구하는 과정은 IC의 Offset Curve(Self Intersection Loop가 없는)상에서 세가지 조건들(Update 조건, Switching 조건, Infeasibility 조건)을 기초로 정의된다. 이 Offset Curve를 이루는 점들은 $Q_i' = Q_i + R \cdot N_i$ 로 구성되며 여기서 N_i' 은 N_i 를 CL평면에 투사시켜 얻은 단위벡터이고 R은 공구반경이다.

A. Update 조건

Q_i' 부터 시작하여 T_1 을 구하는 과정에서 T_1 을 update하기 위한 조건을 정의하기 위해 먼저 Offset Curve의 진행을 나타내는 벡터 V를 설정하면 다음과 같다.(그림 7(a))

$$V = CQ_i' \times N_c' \quad (5)$$

위에서 N_c' 은 공구접촉점에 대한 곡면의 단위법선벡터를 CL 평면에 투사시켜 얻은 단위벡터이다. 그리고 T_L, T_R 을 다음과 같이 정의하되

$$T_L = N_c' \cdot \text{Rot}(V, -90^\circ), T_R = N_c' \cdot \text{Rot}(V, 90^\circ) \quad (6)$$

T_1 의 초기값은 T_L 로 정의된다. Q_i' 를 따라 T_1 을 update하기 위한 조건은 아래와 같다.

$$(T_1 \times CQ_i') \cdot V > 0. \quad (7)$$

만약 Q_i' 에 대해서 위의 조건을 만족하면 T_1 을 CQ_i' 의 단위벡터로 치환한다. T_1 을 구하는 과정과 마찬가지로 T_2 를 구하는 과정에서 제일 처음에 T_2 를 T_R 로 초기화한 후 Q_i' 를 따라 진행하면서 다음조건을 만족하면 T_2 를 치환한다.

$$(T_2 \times CQ_i') \cdot V < 0. \quad (8)$$

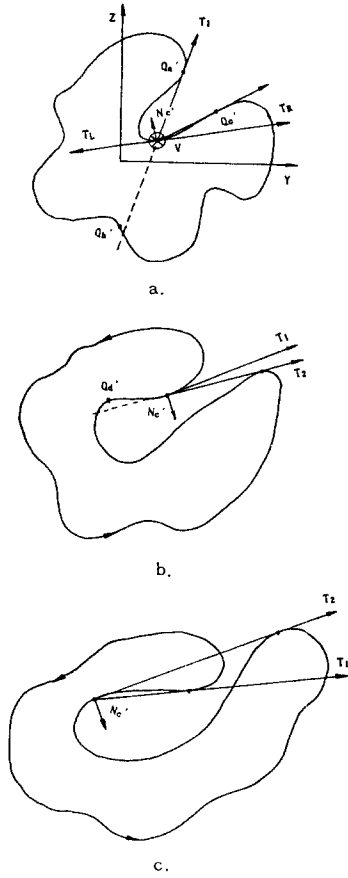


그림 7. T_1, T_2 의 계산

그림7(a)의 Q_a' 에서는 T_1 의 update 조건을 만족하므로 T_1 이 CQ_a' 벡터로 치환되고 Q_c' 에서는 T_2 의 update을 만족하므로 T_2 가 CQ_c' 벡터로 치환된다.

B. Switching 조건

Switching 조건은 T_1 을 구하는 과정을 끝내고 T_2 를 구하는 과정을 시작하기 위한 조건을 의미한다. T_1 의 switching 조건은 아래와 같고 그림7(a)의 경우는 Q_b' 에서 switching 이 일어난다.

$$T_1 \cdot CQ_i' \approx -|T_1| \cdot |CQ_i'| \quad (9)$$

그림7(b)와 같은 곡면의 형상에 대해서는 T_2 를 구하는 과정을 마치기 위한 조건이 필요하다. 이 조건은 다음과 같이 T_1 의 switching 조건과 비슷하다.

$$T_2 \cdot CQ_i' \approx -|T_2| \cdot |CQ_i'| \quad (10)$$

그림7(b)의 Q_a' 는 위조건을 만족하는 점이된다.

c. Infeasibility 조건

그림 7(c)와 같이 부품의 형상에 따라 공구가 접근할 수 없는 CC점이 존재하는 경우가 있다. 따라서 이러한 CC점은 공구경로상에서 제거해야한다.

공구가 접근 불가능한 CC점을 발견하기 위해서는 T_1 과 T_2 를 구한 후에 아래의 조건을 테스트 한다.

$$(T_1 \times T_2) \cdot V < 0. \quad (11)$$

위에서 열거한 조건들을 고려하여 T_1 과 T_2 를 구하는

과정을 정리하면 다음과 같다.

Procedure 3: T_1, T_2 를 구하는 과정

• Step 1: Initialization

1. Read in $Q^i, N^i, i = 1, \dots, n$ and C, N^c .
2. Compute $V = CQ^i \times N^c$.
3. Let $T_1 = N^c \text{ Rot}(V, -90^\circ)$,
 $T_2 = N^c \text{ Rot}(V, 90^\circ)$.
4. $i=0$.

• Step 2: Updating T_1

1. $i = i + 1$
2. If Eq.(9) is satisfied, go to Step 3.
3. If Eq.(7) is satisfied,
then update T_1 by Q^i .
4. Go to 1

• Step 3: Updating T_2

1. If $i = n + 1$, go to Step 4.
2. If Eq.(10) is satisfied, go Step 4.
3. If Eq.(8) is satisfied, then update
 T_2 by Q^i .
4. $i = i + 1$, go to 1.

• Step 4: Termination and Infeasibility check

1. If Eq.(8) holds, remove the corresponding
CC-point from the CC-path.
2. Stop.

3.3 기하모델링방법의 고려

부품의 곡면에 대한 공구경로를 생성할 때 경로상에 존재하는 모든 CC점들에 대하여 제시된 알고리즘을 이용하여 접근가능한 자세를 구하고 간섭을 체크하는 것은 많은 계산 시간을 필요로한다. 따라서 곡면의 형상의 특성을 이용하여 곡면의 일부분만 계산하고 전체곡면에 대한 계산시간을 줄이는 것이 필요하다. [1]에서는 이미 회전형상의 자유곡면을 표현하는 방법들을 제시하였다. 그러므로 여기에서 논의하는 모든내용은 [1]에서 제시된 곡면설계방법을 기초로한다.

A. 한 단면곡선을 스위핑한 곡면

이 곡면의 표현방법은 한 단면(YZ평면과 평행)상에 존재하는 한 폐곡선(closed curve)을 이 단면의 법선방향(X축 방향)으로 스윕(회전이동(rotation)과 선형이동(translation))을 통해 곡면을 표현하는 것이다. 공구경로를 산출할 때에도 이 곡면의 표현 방법의 특성을 이용하면 다음과 같은 계산상의 잇점을 얻을 수 있다. 첫째는 각 CC점에 대한 공구자세를 구하기 위해 IC를 구할 때 IC알고리즘을 이용하지 않고 공구중심점의 x성분을 이용하여 곧바로 구할 수 있다. 그리고 한 단면에 대한 공구경로를 산출하여 스윕(sweep rule)을 따라 전체곡면에 대한 공구경로를 생성하여 계산시간을 많이 줄일 수 있다. w(X축 방향)가 0일 때의 곡선에 대하여 CL데이터인 $CL(u, 0)$ 를 구했을 때 전체곡선에 대한 CL데이터는 다음과 같다.

$$CL(u, w_j) = CL(u, 0) \cdot \text{Rot}(X, \theta(w_j)) \cdot \text{Trans}(X, x(w_j))$$

B. 일반적인 곡면

이 곡면의 형상은 일정한 규칙없이 정의되므로 공구경로를 구할 때에 공구경로상에 존재하는 모든 CC점에 대해 접근가능한 공구자세를 구하고 간섭을 확인해야한다. 그렇게 되

면 모든 공구중심점에 대하여 CL평면을 정의하고 IC를 구해야하므로 많은 계산이 필요하다. 따라서 일반적인 곡면의 공구경로를 산출할 때에 한 w_i 에서 정의되는 각 CC점에 대한 CL평면을 구하여 IC를 구하지 않고, 근사적이지만 모든 CC점에 대한 한 CL평면만을 정의하여 오직 하나의 IC를 구하여 이 곡선상에서 모든 점에 대한 공구자세를 구한다. 위에서 제시한 방법을 따라 공구자세를 구하면 정확하지 않고 근사적이지만 계산시간을 많이 줄일 수 있다.

4. 구현 결과

이 논문에서 제시한 알고리즘은 이미 개발된 4축 CAM 시스템에 구현하였다. 이 CAM시스템은 CAD, CLDATA, GSIM, POST의 4개 모듈로 구성되어 Bridgeport CNC 밀링 머신과 연결되어 있다. 그림8은 여기서 제시한 알고리즘의 유효성을 검증하기 위해 사이드 밀링 커터(side milling cutter)와 같은 형상을 테스트한 결과를 보여주고 있다. 이 부품은 한 단면 곡선을 X축을 따라 길이 50mm 만큼 선형이동 그리고 180°만큼 회전시켜 스윙한 곡면이다. (그림 8(a)) 그림8(b)는 공구반경이 2mm일 때의 공구중심점경로와 선택된 몇개의 CC점에 대한 초기공구자세를 보여주고 있는데 그림상으로 공구자세에 의한 간섭이 존재하는 것을 알 수 있다. 그림8(c)은 공구간섭을 다루는 알고리즘을 사용했을 때의 각 단면에 대한 간섭없는 공구자세를 보여주고 있다. 그림8(d)는 GSIM 모듈의 출력물 스크린카피(screen copy)한 것을 나타내는데 오른쪽 윗부분은 절삭 되어야 할 형상, 왼쪽 윗부분은 공구 끝점의 경로, 그리고 아래 두 그림은 공작물의 회전과 공구의 이동하는 모습을 보여주고 있다. 그리고 그림8(e)는 실제로 시험 절삭한 형상의 모습을 보여주고 있다.

5. 결론 및 요약

본 논문에서는 4축 밀링가공에서 발생하는 공구간섭을 3가지 유형(SSI, TSI, TAI)들을 원인별로 분류하고 이러한 간섭들을 제거하여 공구경로를 생성하는 알고리즘을 제시했다. 첫 두가지 유형의 간섭은 3축이나 4축밀링가공시 모두 발생하지만 마지막 세번째 유형은 4축이상의 밀링가공에서만 생기는 간섭으로 공구경로산출시 매우 중요한 고려사항이다. SSI를 다루는 알고리즘은 간섭이 없게 적당한 공구간격을 결정하는 방법을 제시하였고 TSI에 대해서는 CC평면을 기초로 공구의 크기에 의한 간섭을 확인하는 알고리즘을, 그리고 TAI에 대해서는 CL평면을 기초로 간섭없는 공구자세를 구하는 방법을 보였다. 또한 제시된 알고리즘을 컴퓨터에 구현하여 부품의 곡면에 대한 간섭없는 공구경로를 산출하고 그래픽 시뮬레이션과 실제의 시험절삭을 통하여 그 유효성을 검증하였다. 그리고 제시된 알고리즘은 이미 개발된 4축 CAM 시스템에 확장, 적용하여 존재하고 있는 CAD/CAM 시스템에 구현될 수 있음을 확인하였다.

6. 참고문헌

[1] Suh, S H and Lee, K S 'A prototype CAM system for four-axis NC machining rotationally-free-surface' *J Manufacturing Systems* Vol 10 No 5 (1991)

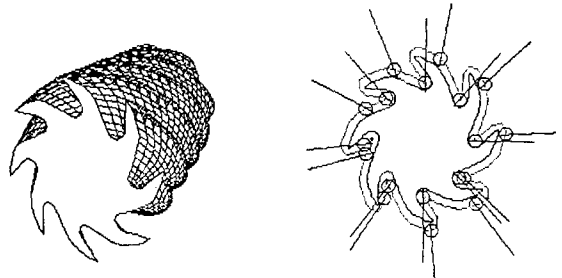
[2] Hansen, A and Arbab F 'Fixed-axis tool positioning with built-in global interference checking for NC path generation' *IEEE J. Robot. Automat.* Vol 4 No 6 (December 1988) pp 610-621

[3] Lee, A C, Chen, D P, and Lin, C L, 'A CAD/CAM system from 3D coordinate measuring data' *Int. J. Prod. Res.* Vol 28 No 12(1990) pp127-136

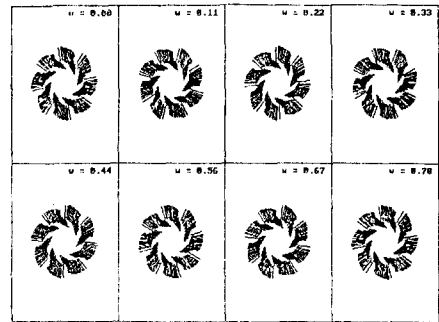
[4] Choi, B K and Jun, C S 'Ball-end cutter interference avoidance in NC machining of sculptured surfaces' *Comput.-Aided Des.* Vol 28 No 12(1989) pp371-378

[5] Chen, Y J and Ravani, B 'Offset surface generation and contouring in computer-aided design' *ASME J. Mech. Trans. Automat. Des.* Vol 109(1987) pp 133-142

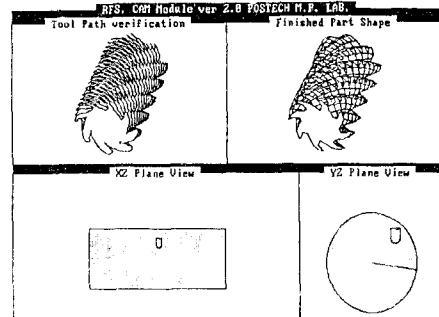
[6] Mortenson, *Geometric modeling* John Wiley & Sons, USA (1985).



a. 부품의 형상 (L=50mm, $\theta=180^\circ$) b. 초기공구자세



c. 8개의 단면에 대한 공구자세



d. 그래픽시뮬레이션



e. 절삭된 형상

그림 8. 구현 결과