삼각망 곡면의 3축 라운드엔드밀 가공을 위한 공구경로 생성

Toolpath Generation for Three-axis Round-end Milling of Triangular Mesh Surfaces

✉정연찬[1]
✉YunChan CHUNG[1]

1 서울산업대학교 금형설계학과 (Department of Die and Mould Engineering, Seoul National Univ. Of Technology)
✉ Corresponding author: ychung@snut.ac.kr, Tel: 02-970-6395

Manuscript received: 2008.8.20 / Revised: 2008.10.23 / Accepted: 2008.11.13

Presented in this paper is a method to generate round-endmill toolpaths for sculptured surfaces represented as a triangular mesh model. The proposed method is applicable in toolpath generation for ball-endmills and flat-endmills because the round-endmill is a generalized tool in three-axis NC (numerical control) milling. The method uses a wireframe model as the offset model that represents a cutter location surface. Since wireframe models are relatively simple and fast to calculate, the proposed method can process large models and keep high precision. Intersection points with the wireframe offset model and a tool guide plane are calculated, and intersection curves are constructed by tracing the intersection points. The final step of the method is extracting regular curves from the intersection curves including degenerate and self-intersected segments. The proposed method is implemented and tested, and a practical example is presented.

Key Words: Round-endmill (라운드엔드밀), Toolpath (공구 경로), Triangular Mesh (삼각망), Offset (오프셋)

기호설명

Rf = flat radius of round-endmill
Rc = corner radius of round-endmill
Tc = tool center point
TOffset = tool offset function

1. 서론

본 연구에서는 자유 곡면을 라운드엔드밀 공구로 3축 NC 밀링 가공할 때 사용되는 공구 경로의 계산 방법을 제안한다. CAD(Computer-Aided Design) 시스템에서 자유 곡면은 매개변수 형태로 표현되지만 매개변수 곡면을 원하는 정밀도의 삼각망(triangular mesh)으로 변환하는 방법은 Piegl[1] 등의 연구자에 의해 잘 정립되어있다.[2] 따라서 본 연구에서는 삼각망으로 표현된 자유 곡면을 가공 대상으로 한다. 삼각망은 3축 NC(numerical control)밀링의 가공 경로 계산에 널리 쓰이고 있다. Flutter[2] 의 연구에서는 가공 경로 계산 방법의 최근 변화로 삼각망 모델의 사용을 지적하고 있다. 이미 1970년대에 Mair 와 Duncan 은 삼각망 모델에 근거한 가공 경로 계산을 시도하였지만 당시에는 컴퓨터의 용량과 성능의 문제로 한계가 있었다.[3] 일반적인 금형 형상의 경우 원하는 정밀도로 곡면을 근사하면 삼각형의 개수가 수십만 혹은 수백만 개에 이르기 때문이다. 최근 컴퓨터 기억 용량과 계산 성능의 비약적인 발전으로 삼각망 모델을 이용한 방법이 새롭게 부각되고 있다. 그리고 컴퓨터 그래픽스, 3차원 형상 측정과 측정 데이터를 이용한

역설계, 쾌속 조형 등의 분야에서도 삼각망이 다양한 용도로 쓰이고 있다.[4]

본 연구의 공구 경로 계산에 사용되는 삼각망은 위상 정보를 가지고 있어서 정점을 어떤 삼각형이 공유하는지 알 수 있으며, 모서리를 공유하는 두 삼각형도 알 수 있다. 널리 쓰이는 STL 파일 형식은 이러한 위상 정보가 없다. 그러나 위상 정보가 없는 삼각형 집합에서 위상 정보를 부여하는 방법에 관한 연구도 잘 정립된 분야이다.[5] 그리고 하나의 자유 곡면을 삼각망으로 변환하는 경우는 대개 위상 정보를 유지할 수 있다.

3 축 밀링 가공 경로 계산의 기존 연구 [6-9] 들은 대개 볼엔드밀 공구를 고려하지만 본 연구는 라운드엔드밀의 공구 경로 계산을 연구 대상으로 한다. 라운드엔드밀은 Fig. 1 에서 보듯이 자유 곡면의 가공에 널리 쓰이는 세 가지 공구의 일반형이다. 볼엔드밀은 라운드엔드밀의 평반경(flat radius) Rf 가 0 인 특수 경우이고, 평엔드밀은 라운드엔드밀의 코너반경(corner radius) Rc 가 0 인 특수 경우다. 따라서 라운드엔드밀을 위한 공구 경로 계산 방법은 볼엔드밀과 평엔드밀 공구 경로 계산에도 사용할 수 있다. 그리고 최근 라운드엔드밀 공구의 사용이 늘고 있다. 라운드엔드밀은 볼엔드밀에 비해 가공 생산성이 높고 인서트 팁 형태의 공구가 일반화되면서 황삭과 중삭은 물론이고 정삭에도 라운드엔드밀 공구가 사용되기 때문이다.[10]

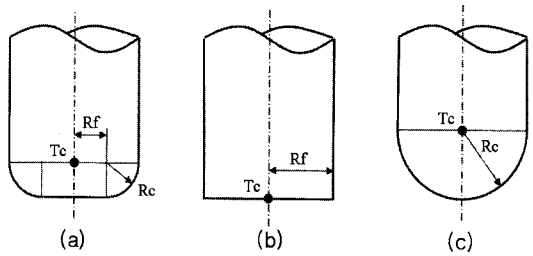


Fig. 1 (a) round-endmill, (b) flat-endmill (c) ball-endmill (Rf > 0 and Rc > 0)

2. 기존 연구

삼각망에 기초한 3 축 밀링의 공구 경로 계산은 Fig. 2 에 정리한 것처럼 점 기반 방식과 곡선 기반 방식으로 나뉜다.[11] 점 기반 방식은 주어진 (x, y)위치에서 공구를 수직 방향으로 내릴 때 공구와 삼각망이 최초로 닿는 공구 위치점(cutter location point: CL point)을 찾는 방법이다.[4] 이 방식은 공구 형상에 따른 제약이 적고 계산 알고리즘이 단순하지만 계산량이 많다는 단점이 있다. 이 방식은 공구 경로를 연속적인 곡선으로 생성하지 못하고 공구 경로에 놓이는 점들을 짧은 간격으로 샘플링 해야 한다. 날카로운 모서리의 경우 정밀도를 높이기 위해 더욱 많은 샘플링이 필요하고 그 만큼 계산량이 증가한다. Hwang,[7] Flutter[2] 등의 연구는 계산량을 줄이기 위한 효율적 방법을 제시하고 있다.

곡선 기반 방식은 공구 위치 곡면(cutter location surface)을 생성한 후 그 곡면에 경로를 투영해서 공구 경로를 얻는다. 이 방식은 정밀한 공구 경로를 얻을 수 있지만 공구 위치 곡면을 생성하기가 쉽지 않다. 공구 위치 곡면은 공구의 형상을 고려한 곡면의 오프셋 모델이다. Jun[8] 은 볼엔드밀 공구에 한정해서 삼각형과 모서리, 정점의 오프셋 모델을 각각 삼각형과 원통면(cylindric surface), 구면(spherical surface)으로 표현하고, 그 오프셋 모델을 경로 안내 평면으로 잘라서 공구 경로를 얻었다.

이 방법은 볼엔드밀의 경우에만 적용할 수 있다. Kim[11] 은 공구의 일반형인 APT 공구의 공구 위치점 계산 방법을 제시하고, 삼각형으로 이루어진 오프셋 모델을 생성하는 방법을 제시했다. 그의 방법은 모서리와 정점의 오프셋 정밀도를 높이면 오프셋 모델의 삼각형 개수가 많아져 모델의 메모리 용량이 커지는 단점이 있다. Chung[4] 은 오프셋 모델과 경로 안내 평면의 교선을 효율적으로 계산하기 위해 교선을 추적하는 방법을 제시하였다. 오프셋 모델이 비정규 삼각망인 경우 오프셋 면을 경로 안내 평면으로 절단하면 그 결과 곡선에는

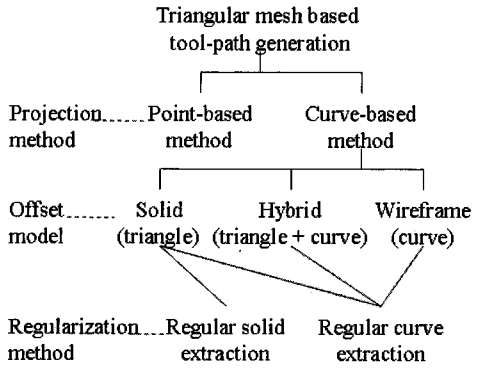


Fig. 2 Technology hierarchy of tool-path generation

꼬임과 불필요한 곡선이 존재한다. Park[9] 은 이러한 곡선의 꼬임과 불필요한 곡선을 효율적으로 제거하는 방법을 제시하였다. Shin 등은 솔리드(watertight solid) 삼각망의 경우에 꼬임이 없는 정규 삼각망을 얻는 방법을 제시하였다.[12]

본 연구는 기존 연구와 달리 공구 경로 계산에 사용하는 오프셋 모델로 와이어프레임 모델(wireframe model)을 사용한다. 그리고 볼엔드밀과 평엔드밀의 일반형인 라운드엔드밀을 대상으로 한다.

3. 오프셋 모델 생성

3.1 라운드엔드밀의 공구 위치점

공구 경로 계산을 위한 오프셋 모델은 곡면 위의 모든 점에 대응하는 공구 위치점의 집합으로 정의된다. Fig. 3 은 라운드엔드밀의 공구 위치점 계산을 도식적으로 보여준다. 곡면 위의 한 점 P 에 접촉하는 공구 위치점 Tc 는 그 점의 단위 곡면 법선 벡터 N 과 공구 정의 변수 Rf, Rc 로 계산된다.[13] 특정 공구의 오프셋 모델 계산에서 공구 정의 변수 Rf, Rc 는 상수이므로 공구 위치점 계산식은 식 (1)과 같이 공구 접촉점과 단위 법선 벡터의 함수로 표현된다. 식에서 Rc 가 0 이면 평엔드밀에 해당하고 Rf 가 0 이면 볼엔드밀에 해당함을 알 수 있다. 식에서 벡터 M 은 법선 벡터 N 에 의해 정의되며, N 의 수평 성분 단위 벡터다. N 의 수평 성분이 0 인 경우는 M 을 0 벡터로 취급해야한다.

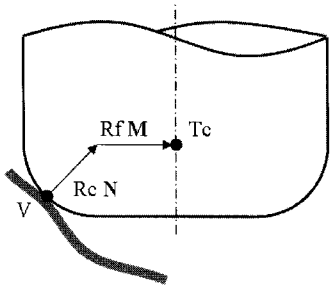


Fig. 3 Computation of cutter location

$$Tc = TOffset(P, N) = P + Rc\,N + Rf\,M \quad (1)$$

where $M = (Nx, Ny, 0) / |(Nx, Ny,, 0)|$

3.2 삼각형의 오프셋

삼각형의 오프셋은 삼각형을 구성하는 세 정점을 공구 위치점으로 오프셋해서 얻는다. 세 정점을 잇는 직선 와이어로 오프셋 삼각형이 정의된다. 정점의 오프셋에 사용되는 법선 벡터는 삼각형의 법선 벡터를 사용한다. 결과적으로 원래의 삼각형 T 를 (Rc N + Rf M)만큼 단순 이동 변환한 결과가 오프셋 삼각형 OT 다(Fig. 4 참조). 이 방법은 기존의 다른 연구 [4,8,9,11] 와 개념적으로 동일한 방법이다.

3.3 모서리(edge)의 오프셋

각각의 삼각형을 공구 위치점으로 오프셋하면 원래 잘 맞아있던 모서리가 벌어지거나 겹쳐진다. 오프셋해서 겹쳐지는 오목 모서리는 공구 위치 곡면에 영향을 미치지 못하기 때문에 볼록 모서리만 고려하면 된다. Kim[11] 의 연구에서는 볼록 모서리의 오프셋 모델을 삼각망으로 표현하였다. 본 연구에서는 모서리의 오프셋 모델로 네 개의 경계 곡선으로 표현되는 와이어프레임 모델을 사용한다. 네 경계에서 삼각형과 경계를 이루는 부분은 오프셋 삼각형의 모서리이므로 직선이다. 그러나 라운드엔드밀의 경우 다른 두 경계 곡선은 함수식으로 표현이 곤란하므로 점열(point sequence) 곡선으로 표현한다. Jun[8] 의 연구에서는 볼엔드밀의 경우에 2 차 유리 베지어(quadratic rational Bezier)로 곡선 부분을 표현하였다. 모서리의 끝점에 대응하는 점열 곡선은 모서리의 시작점에 대응하는 점열 곡선을 모서리를 따라 단순 이동 변환해서 얻어진다. 오프셋에 사용되는 두 개의 벡터(N1, N2)가 동일하기 때문이다. 결국 모서리의 오프셋 면은 두 점열 곡선의 룰드 곡면(ruled surface)으로 해석된다. Fig. 5 에서 정점 V1 에 대응하는 점열 곡선을 만드는 절차는 다음과 같다.

Step 0. 모서리를 이루는 두 삼각형의 법선 벡

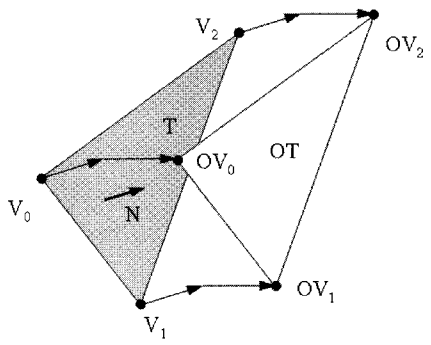


Fig. 4 Offset of a triangle

터로 두 개의 공구 위치점 Ps, Pe 를 계산하고 두 점으로 이루어진 최초의 점열 곡선을 생성한다. N1 과 N2 는 모서리에 인접한 두 삼각형의 단위법선 벡터다.

$$P(0) = Ps = TOffset(V1, N1)$$
$$P(1) = Pe = TOffset(V1, N2)$$

점열 곡선의 모든 인접한 두 점이 근사 오차를 만족할 때까지 다음의 절차를 반복하면서 점을 삽입한다. 최초의 i 는 0 이며 P(i)는 점열 곡선의 i 번째 점이다. N(i)는 공구 위치점 P(i) 계산에 사용된 단위벡터다.

Step 1. 인접한 두 점 P(i)와 P(i+1) 사이에 삽입할 새로운 공구 위치점 Pn 을 구한다.

$$Nn = (N(i) + N(i+1)) / |(N(i) + N(i+1))|$$
$$Pn = TOffset(V1, Nn)$$

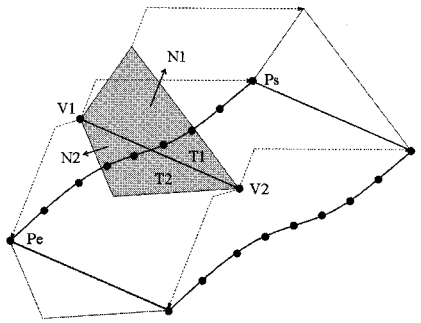


Fig. 5 Offset of an edge

Step 2. 새로 계산된 Pn 을 이용해서 인접한 두 점 P(i)와 P(i+1)의 근사 오차 e 를 계산한다.

e = P(i)와 P(i+1)을 잇는 선분과 Pn 의 수직 거리

Step 3. 새로운 점 Pn 의 삽입 여부를 판단한다.

if (근사 오차 e 가 허용 정밀도를 만족)

    i 의 값을 하나 증가

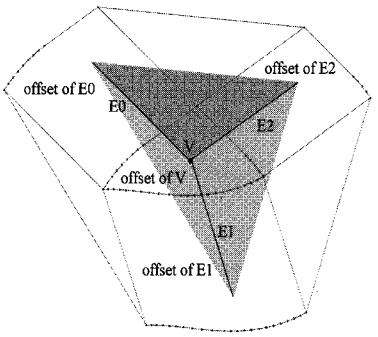
else

    P(i)와 P(i+1) 사이에 Pn 을 삽입

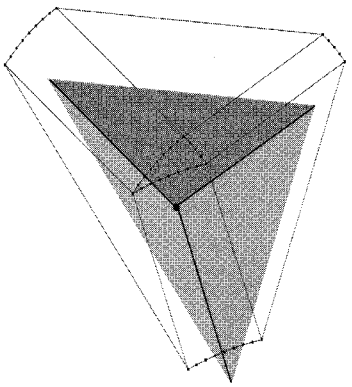
Step 1 으로 돌아가서 절차를 반복한다.

3.4 정점(vertex)의 오프셋

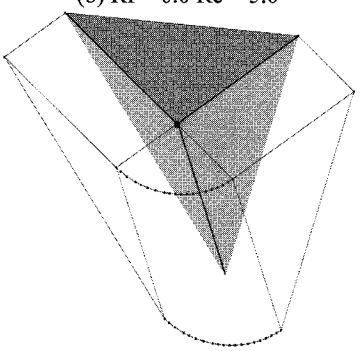
정점의 오프셋 결과는 주변 모서리의 오프셋



(a) Rf = 5.0 Rc = 5.0



(b) Rf = 0.0 Rc = 5.0



(c) Rf = 5.0 Rc = 0.0

Fig. 6 Wireframe offset model

면을 이루는 경계 곡선으로 표현된다. 따라서 별도의 계산이 필요 없다. 경계 곡선의 개수는 주변 볼록 모서리의 개수와 같다. Fig. 6 에서 가운데 세 개의 경계 곡선으로 제한되는 부분이 정점 V 의 오프셋 면에 해당한다.

4. 오프셋 모델의 교선과 경로 곡선 생성

오프셋 모델은 공구 중심의 이동 궤적면(swept

surface)이므로 오프셋 모델에 투영되는 곡선은 공구 경로에 해당한다. 꼬임이 제거된 정규 오프셋 모델의 경우는 수직 평면과의 교선이 수직 방향 투영으로 얻어지는 곡선과 동일하다. 꼬임이 제거되지 않은 오프셋 모델의 경우에는 교선이 꼬임을 포함하고 있어서 정규화 과정을 거쳐야 최종 공구 경로로 사용할 수 있다.[4] 본 연구의 오프셋 모델은 꼬임이 제거되지 않은 모델이므로 교선을 얻은 후 곡선의 정규화 과정을 통해 꼬임을 제거한다.

4.1 오프셋 면과 수직 평면의 교점 계산

본 연구의 오프셋 모델은 와이어프레임 모델이므로 먼저 와이어프레임과 교점을 구하고 그 교점을 이어서 교선을 얻는다. 오프셋 모델을 구성하는 모든 와이어 는 하나의 직선 선분이거나 선분의 연속이다. 따라서 오프셋 모델과 수직 평면의 교점 계산은 직선과 평면의 교점 계산으로 단순화된다. 직선과 평면의 교점 계산은 매우 간단하며 안정적인 구현이 가능하다.[14]

삼각형의 오프셋 면은 3 개의 직선 와이어로 구성되며 교점이 존재하는 경우에는 일반적으로 2 개의 교점이 생성된다. 교점이 1 개인 경우와 모서리에서 접하는 경우 등은 별도로 처리한다. Fig. 7 에서 P0, P1, P2, P5, P6 등이 삼각형의 오프셋 면을 구성하는 와이어와의 교점이다.

한 모서리의 오프셋 면을 이루는 4 개의 와이어 중에 2 개의 직선 와이어는 삼각형 오프셋 면 정의에 사용되었다. 따라서 2 개의 곡선 와이어와 평면의 교점을 계산한다. 곡선 와이어는 선분의 연속으로 정의되므로 선분과 평면의 교점 계산을 반복한다. Fig. 7 에서 P3 과 P4 가 모서리에 대응하는 곡선 와이어와 평면의 교점이다. 교점이 존재하는 경우 모서리의 오프셋 면과 평면의 교점은 Fig 8 과 같이 2 개 혹은 4 개가 생성된다.

정점의 오프셋 면은 주변 모서리의 오프셋 와이어로 구성된다. 따라서 별도의 교점 계산이 필요 없다. 즉, Fig. 7 에서 P3, P4 는 이미 계산된 교점이다. 그러나 앞서 구해진 교점들이 어떤 오프셋 와이어에 의해 생성된 것인지 알 수 있어야 한다. 임의의 정점에 해당하는 모든 교점 혹은 임의의 모서리에 해당하는 모든 교점에 접근할 수 있도록 자료 구조를 구성했다.

4.2 오프셋 면과 평면의 교선 계산

4.1 절에서 얻어진 교점을 연결하여 교선을 계산한다. 삼각형의 오프셋 면은 평면이므로 교선은 직선이다. 따라서 Fig. 7 의 (P0, P1), (P1, P2), (P5, P6) 은 각각 직선으로 연결한다.

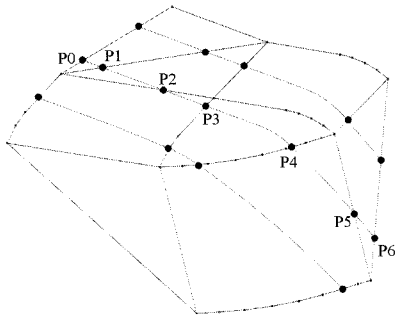


Fig. 7 Intersection points and curves

Fig. 7 에서 (P2, P3)와 (P4, P5)는 모서리의 오프셋 면을 지나는 교선이다. 모서리의 오프셋 면을 지나는 교선은 Fig. 8 과 같이 다양하다. 그러나 모서리 오프셋 면은 두 개의 동일한 모양의 점열 곡선을 직선으로 연결한 룰드 곡면이므로 직선과 평면의 교점 계산을 반복하면 두 교점을 잇는 교선을 얻을 수 있다.

두 교점이 정점의 오프셋 면을 지나는 경우는 정밀도를 만족할 때까지 두 교점의 내부에 점을 삽입해서 정밀한 교선을 얻는다. Fig. 9 와 같이 정점 V 에 역공구(inverse tool)의 중심을 두면 두 교점 P1 과 P2 는 역공구의 표면에 존재하고 두 교점을 잇는 교선은 역공구를 절단해서 얻을 수 있다.

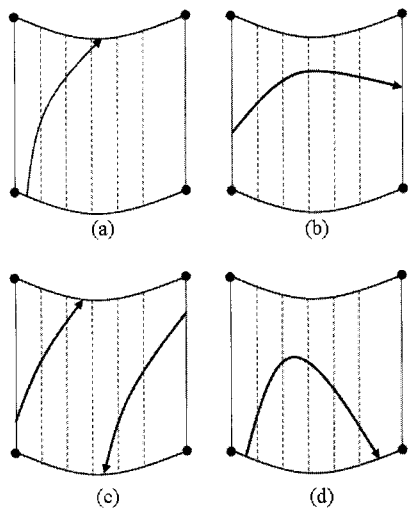


Fig. 8 Intersection with offset of an edge

이때의 교선도 곡선식으로 얻는 것이 곤란하기 때문에 3.3 절에서 설명한 모서리 오프셋의 점열 계산과 같은 반복적인 방법으로 점열을 계산한다.

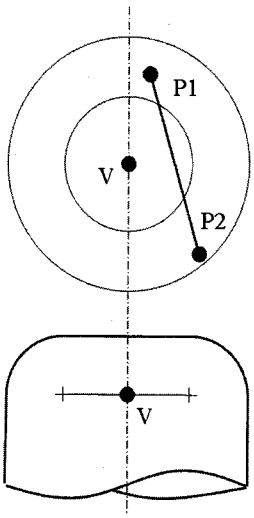
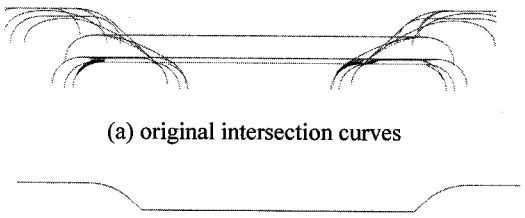


Fig. 9 Slicing offset of a vertex



(a) original intersection curves

(b) regularized curve

Fig. 10 Regularization of intersection curves

4.3 교선의 추적과 연결

오프셋 면을 옮겨 다니면서 4.2 절에서 구한 교선을 길게 연결한다. 인접한 오프셋 면이 없거나 오목한 모서리를 만나면 연결을 중단한다. Fig. 7 에서 (P0, P1) 교선의 끝인 P1 에서 (P1, P2) 교선을 추적 연결한다. 같은 방법으로 P3, P4, P5, P6 을 따라 교선을 연결한다. P6 과 같이 더 이상 인접한 오프셋 면이 없거나 오목 모서리를 만나면 교선의 추적을 중단한다.

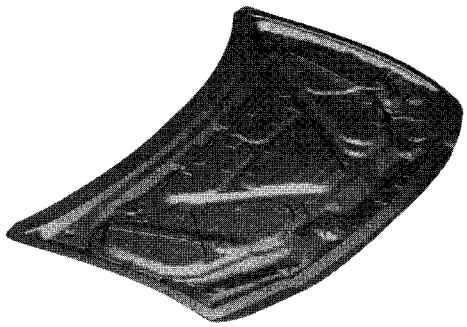
4.4 교선의 추적과 연결

본 연구에서는 꼬임이 제거되지 않은 오프셋 모델을 사용한다. 4.3 절에서 얻어진 교선은 Fig. 10(a)에서 보는 것과 같이 서로 교차하는 부분과 불필요한 부분이 존재한다. 따라서 꼬임을 제거하고 가공에 필요한 부분만 취해서 (b)와 같이 정규화된 교선을 얻는 과정이 필요하다. 본 연구는 3 축 가공에 한정하므로 z 값이 큰 교선을 취하면 된다. 복잡하게 얽힌 교선에서 공구 경로로 사용할 수 있는 교선을 찾는 문제는 개념적으로 간단하지만 실제적인 구현에는 계산 효율성과 안정성의 문제가 대두된다. 본 연구는 Park[9] 의 방법을 사용하였다.

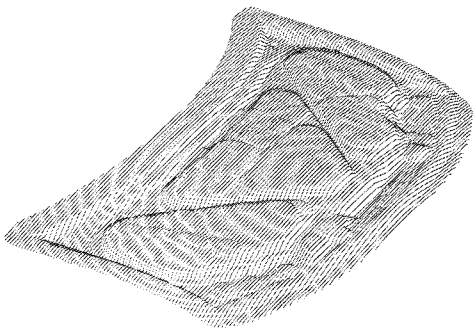
5. 구현 결과

제안된 방법은 마이크로소프트사의 Visual C++ 로 구현되었으며 다양한 예제로 테스트 하였다. 생성된 공구 경로는 NC 모의 가공 시스템을 통해 검증했으며 실제 가공에도 문제가 발견되지 않았다. 구현 결과는 상용 CAM 시스템에서 사용 중이다.

Fig. 11(a)는 가공 예제 형상이며 자동차 후드 내판이다. 예제 형상의 평면 크기는 대략 1,450 x



(a) Triangular mesh (1,266,345 triangles)



(b) Tool-path

Fig. 11 Example part and tool-path

1,000mm^2 다. 형상 정밀도를 유지하기위해 가공 형상은 약 127 만개의 삼각형으로 근사되었다. Fig. 11(b)는 예제 형상을 Rc=5mm, Rf=10mm 인 공구로 정밀도 0.001mm 로 계산한 결과다. 그림의 이해를 돕기 위해 경로 간격을 10mm 로 했다.

Table 1 은 Fig. 11 의 예제를 주사선 경로 간격 1mm 로 공구 경로를 계산할 때 소요되는 계산 시간이다. STL 파일을 읽고, 초기 삼각망 모델을 구성하는데 15 초 걸렸다. 이때 사용된 컴퓨터 메모리는 대략 50MB 다. 초기 삼각망을 입력으로 오프셋 모델인 와이어프레임 모델 생성에 62 초가 소요되었다. 모서리의 오프셋 면을 이루는 곡선 와이어 생성에 많은 시간이 소요됨을 알 수 있다. 오프셋 모델 생성에 사용된 메모리는 대략 1GB 다. 교선 계산과 교선의 정규화에 걸리는 시간은 가공 경로의 길이와 개수에 비례한다. Table 1 의 소요 시간은 Pentium 4 CPU 3.2 GHz, 2GB RAM, Windows XP 에서 실행한 결과다.

Table 1 Computation time for an example

step	computation time
STL reading & model construction	15 sec
offset model generation	62 sec
intersection & regularization	21 sec
total	98 sec

6. 결론

라운드엔드밀을 사용하는 3 축 밀링 가공의 공구 경로 계산 방법을 제안하였다. 제안된 방법은 삼각망으로 표현된 곡면을 입력으로 하며 와이어프레임 모델을 오프셋 모델로 사용하였다. 와이어프레임 모델은 개념적으로 다른 모델에 비해 상대적으로 적은 메모리를 요구한다. 결과적으로 제안된 방법은 적은 메모리로 계산을 수행할 수 있어서 많은 개수의 삼각형으로 구성된 곡면을 높은 정밀도로 계산할 수 있었다.

제안된 방법과 다른 방법 혹은 다른 오프셋 모델을 사용한 공구 경로 계산 방법과 정량적인 계산 시간 비교 및 메모리 사용량의 비교 연구가 좋은 후속 연구가 될 것이다. 제안된 방법은 3 축 밀링의 주사선 가공으로 제한된다. 등고선 가공과 잔삭 가공 등의 다양한 가공 방식을 지원할 수 있는 방안의 연구가 필요하다.

참고문헌

1. Piegl, L. A. and Richard, A. M., "Tessellating trimmed NURBS surfaces," Computer-Aided Design, Vol. 27, No. 1, pp. 6-26, 1995.
2. Flutter, A. and Todd, J., "A Machininig Strategy for Toolmaking," Compuer-Aided Design, Vol. 33, No. 13, pp. 1009-1022, 2001.
3. Mair, S. G. and Duncan, J. P., "Polyhedral NC program documentation," Report, Department of Mechanical Engineering, University of British Columbia, 1975.
4. Chung, Y. C., "Tool-path Computing by Slicing Offset Triangles and Tracing Intersections," Transactions of the SCCE, Vol. 10, No. 6, pp. 455-464, 2005.
5. Shin, H., Park, J. C., Choi, B. K., Chung, Y. C. and Rhee, S., "Efficient Topology Construction from Triangle Soup," Proc. of Geometric Modeling and Processing, pp. 359-364, 2004.
6. Hansen, A. and Arbab, F., "Fixed-axis tool positioning with built-in global interference checking for nc path generation," IEEE Journal of robotics and automation, Vol. 4, No. 6, pp. 610-621, 1988.
7. Hwang, J. S. and Chang, T. C., "Three-axis machining of compound surfaces using flat and filleted endmills," Computer-Aided Design, Vol. 30, No. 8, pp. 641-647, 1998.
8. Jun, C. S., Kim, D. S. and Park, S., "A new curve-based approach to polyhedral machining," Computer-Aided Design, Vol. 34, No. 5, pp. 379-389, 2002.
9. Park, S. C., "Sculptured surface machining using triangular mesh slicing," Computer-Aided Design, Vol. 36, No. 3, pp. 279-288, 2004.
10. Sandvik Coromant, "Application Guide: Die & Mould Making," Sandvik Coromant, pp. 41-42, 1999.
11. Kim, S. J. and Yang, M. Y., "Triangular mesh offset for generalized cutter," Computer-Aided Design, Vol. 37, No. 10, pp. 999-1014, 2005.
12. Jung, W. H., Jeong, C. S., Shin, H. Y. and Choi, B. K., "Extracting a Regular Triangular Net for Offsetting," Transactions of the SCCE, Vol 9, No. 3, pp. 203-211, 2004.

13. Choi, B. K. and Jerard, R. B., "Sculptured Surface Machining," Kluwer Academic Publishers, pp. 41-43, 1998.
14. O'Rourke, J., "Computational Geometry in C," Cambridge University Press, pp. 220-285, 1998.