

<論 文>

최대경사방향 트리를 이용한 삼각형요소화 곡면모델의 NC 엔드밀링가공에 관한 연구

맹 희 영*

(1994년 11월 8일 접수)

NC End Milling Strategy of Triangulation-Based Curved Surface Model Using Steepest Directed Tree

Hee-Young Maeng

Key Words : Steepest Directed Tree(최대경사방향 트리), Triangulation-Based Curved Surface Model(삼각형요소화 곡면모델), NC End Milling(NC 엔드밀링가공), Cutter Path(절삭경로), Machining Oriented Feature(절삭지향적 특징)

Abstract

A novel and efficient cutter path planning method for machining intricately shaped curved surfaces, called the steepest directed tree method, is presented. The curved surface is defined by triangular facets, the density and structure of which are determined by the intricacy and form accuracy of the surface. Geometrical form definition and recognition of the topological features are used to connect the nodes of the triangulated surface meshes for the successive and interconnected steepest pathways, which makes good use of end milling characteristics. The planetary cutter centers are determined to locate along smoothly changing paths and then the height values of the cutter are adjusted to avoid surface interference. Several machined examples of intersecting and intricate surfaces are presented to illustrate the benefits of the new approach. It is shown that due to more consistent geometry matching between cutter and surface(in comparison with the current CC Cartesian method) surface finish can be typically improved. Moreover, the material in concave fillets which is difficult to be removed by ball mills can be removed efficiently. The built-in positioning of cutter to avoid interference runs minutely in the sharp and discontinuous regions. The steepest upward movement of the cutter gives a stable dynamic cutting state and allows increase in the feedrate and spindle speed while remaining the stable cutting state.

1. 서 론

최근 많은 공업제품들에서 심미적인 형상 또는

기능곡면의 수요가 급증함에 따라 이들 곡면형을 효과적으로 NC가공할 수 있는 방법에 대한 요구도가 높아지고 있다. 복잡한 곡면을 이상적으로 밀링가공할 수 있는 절삭경로라면 무엇보다도 표면거칠기가 균일하면서 굽도록 조절되어야 할 것이며 언더컷이나 오버컷을 발생시키지 않으면서 가공중

*정회원, 서울산업대학교 기계설계학과

안정된 절삭상태가 유지되도록 하여 절삭조건을 높일 수 있고 가공시간을 단축시킬 수 있어야 할 것이다.

현재 스플라인 모델에 기초를 두고서 상용적으로 사용되고 있는 절삭경로의 산출방법으로는 크게 다음과 같은 3가지 방법으로 나눌 수 있다.^(1~6) Iso-parametric 방법에서는 곡면위의 일정한 파라미터 값을 갖는 곡선을 구하여 이 곡선위에 커터와 곡면과의 접촉점이 위치하도록 산출하며, CC-Cartesian 방법에서는 곡면과 커터의 운동을 안내하는 구동면(drive plane)들과의 교차곡선을 구하여 이를 접촉점으로 사용하고, CL-Cartesian 방법에서는 곡면과 구동면에 동시에 접하는 일련의 커터 위치들을 산출하고 있다.

한편 특수한 목적을 위한 절삭경로의 산출방법으로 contour map 방법^(7,8)에서는 황삭가공을 목적으로 곡면의 등고선을 구하여 이를 접촉점으로 이용하고 있고, offset profile 방법^(9,10)에서는 포켓형 곡면의 가공을 목적으로 경계 폐곡선에 등간격으로 오프셋 된 폐곡선들을 따라 안내되면서 커터가 바닥의 곡면에 구속되도록 산출하고 있기도 하다.

그러나 이들 방법들은 대부분 기하학적인 산출방법의 편리성에 바탕을 두고서 개발된 방법들이어서 가공후 커슥(cusp) 크기가 균일하게 생성되지 않을 뿐 아니라 커터의 절삭성을 열악하게 하는 운동형태가 자주 발생하고 있고, 구성부 또는 복합곡면의 교차부위에서 비절삭체적(uncut volume)을 과대하게 남기게 하는 절삭경로들이 산출되고 있다. 그리고 이들 방법들에서는 일반적으로 볼밀(ball mill)이 사용되는 것을 기초로 하고 있으나, 절삭성은 엔드밀(flat end mill)에 비해 일반적으로 좋지 못하다.⁽¹¹⁾ 이의 해결을 위해 한편으로는 절삭능력이 좋은 볼밀공구를 개발하려는 연구들이 이루어져 왔고,^(11,12) 다른 한편으로는 엔드밀을 자유곡면의 가공에 사용하고자 하는 연구⁽¹³⁾가 진행되어 왔으며 5축 가공에서는 커터치짐량과 커슥크기를 최소화할 목적으로 일부 응용하고 있기도 하다.^(14,15)

한편 3축 밀링가공에서 곡면을 엔드밀 커터로 가공을 할 때 표면거칠기를 가장 좋게 나타내고 있는 절삭경로는 곡면의 최대경사방향을 따라서 가공하는 경로임이 해석적으로 고찰되고 있으나^(16,17) 곡면가공에 있어서는 절삭경로 산출상의 복잡성과 산만한 경로들을 적절하게 연결할 수 있는 방법론이 개발되어 있지 못하여서 실용적으로 이를 응용하고

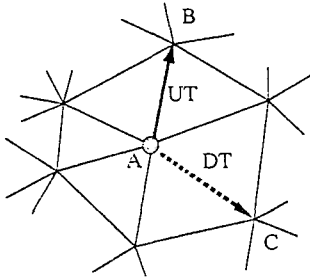
있지 못하고 있다. 또한 엔드밀 커터는 각주형 가공물의 절삭을 대상으로 공구가 설계되어 있어서 바닥면의 절삭날들에 과부하가 생기기 쉬운 플린지 컷 유형의 절삭경로는 절삭성을 열악하게 만드는 주된 원인이 된다.

따라서 본 연구는 삼각형요소로 표현된 곡면모델의 NC가공에서 엔드밀 커터가 곡면의 최대 경사선을 따라 운동하도록 하는 최대경사방향 트리추적법(steepest directed tree method)을 개발하였다. 이 방법은 곡면의 형상공차가 가급적 균일하게 유지되도록 분포되어 있는 삼각형요소화(triangulation)된 곡면모델을 대상으로 하고 있으며, 삼각형요소화된 메쉬들의 방향성과 토폴로지들을 이용하여 방향트리들을 연속적으로 추적할 수 있는 방법론을 사용하고 있다. 그리고 커터가 최대 경사선을 따라 부드럽게 움직이도록 삼각형 모서리들로 연결된 초기의 트리들을 평활화하는 방법론이 고안되었으며, 또한 커터의 각 운동요소마다 곡면형과의 공구간섭(cutter interference)을 효율적으로 체크하는 방법론도 제시되었다. 결과적으로 종합화된 소프트웨어는 복합곡면, 자유곡면, 조각곡면 등의 3가지 곡면유형을 대상으로 적용 성능평가 실험이 실시되어 그 응용효과 및 실용성이 검토되었다.

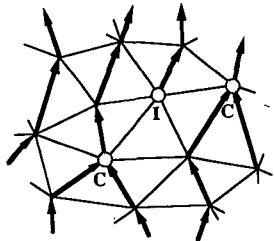
2. 최대경사방향 트리추적법

2.1 기초 개념과 설정

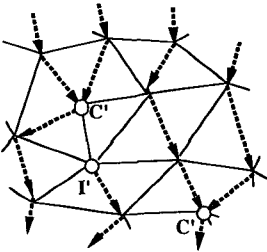
삼각형요소화된 메쉬곡면상의 임의 절점에 대해 이웃의 절점에 대한 최대 경사선은 공간상의 경사도에 의해 결정되고 나무모양의 구조로 연결된다. 최대 경사선은 Fig. 1에서 보이는 바와 같이 Bold(upward tree edge; UT)와 Xold(downward tree edge; DT)의 2종류로 나뉜다. 상향 트리에지(UT)는 수평면의 위로 향하는 방향선중 최대경사의 것(Fig. 1(a)에서 실선으로 표시)으로 정의하며 하향 트리에지(DT)는 수평면의 아래로 향하는 방향선중 최대경사의 것(Fig. 1(a)에서 점선으로 표시)으로 정의한다. 또한 방향선에 대한 절점의 위치에 따라서 UT와 DT는 각각 2가지로 분리하여 취급하는데(UT_0 와 UT_i , 또는 DT_0 와 DT_i), Fig. 1(a)에서 두 최대경사방향선은 A점을 출발하여 각각 이웃의 B점 및 C점으로 향하기 때문에 절점 A의 관점에서 볼 때에는 각각 UT_0 및 DT_0 로 표시되



(a) Upward tree(UT) and downward tree(DT) edges



(b) Steepest upward directed tree



(c) Steepest downward directed tree

Fig. 1 Configuration of steepest upward and downward directed tree

고, 반대로 절점 B의 관점에서 볼 때에는 UT_i 그리고 절점 C의 관점에서 볼 때에는 DT_i 로 표시된다. Fig. 1(b)는 이들 정의에 의한 UT 들로써 구성된 최대경사 상향 트리(steepest upward directed tree) 구조를, 그리고 Fig. 1(c)는 DT 들에 의해 구성된 최대경사 하향 트리(steepest downward directed tree) 구조를 보이고 있다.

이들 최대경사방향 트리들의 특성은 다음의 5가지 정리(propositions)들을 도출하고 6가지 기하형태(geometric form)들을 정의함으로써 그 성질이 파악될 수 있다.

2.1.1 정리 1

유한한 삼각형요소화된 메쉬내의 임의의 절점에 대해서 UT_i (또는 DT_i)는 여러개 존재할 수 있으나

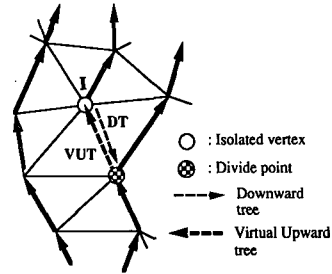


Fig. 2 Isolated vertex and Virtual UT_0 resulting a divide point

UT_0 (또는 DT_0)는 1개 이하(1 또는 0)만 존재한다. 그리고 모든 절점들에서 결정된 UT_0 들을 합한 개수는 모든 절점들에서 결정된 UT_i 들을 합한 개수와 같으며, 이는 DT_0 와 DT_i 의 관계에서도 동일하게 성립된다.

Fig. 2의 예에 있는 절점 I와 같이 UT_i 가 없는 점은 고립점(isolated vertex)으로 명명하며, 이같은 고립점에서는 실제의 UT_i 대신에 이 점에 대한 DT_0 의 역 방향으로 정의되는 가상 UT_i (virtual UT_i ; VUT_i)를 설정할 수 있다. 한편 고립점에 대한 VUT_i 의 상대점(opposite vertex)에서는 이 방향선이 VUT_0 로 설정되며 정상적인 UT_0 와 함께 최대경사 상향 트리의 경로가 분기되는 현상을 발생시킨다.

2.1.2 정리 2



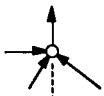


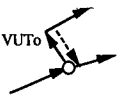

최대경사 상향 트리에서의 임의의 고립점에 대하여, 이 고립점은 한번의 VUT_i 에 의한 연결로 정상적인 UT_0 를 갖는 절점에 연결될 수 있고, 이 연결된 상대점은 항상 분기점(divide point)의 역할을 담당한다.

곡면 위의 임의의 절점이 곡면형상의 변화과정에서 어떠한 역할을 차지하고 있는가를 지정하기 위해서는 Table 1과 같이 정점(apex), 합치점(combine), 보통점(normal), 능선점(ridge), 함몰점(sink) 그리고 계곡점(valley) 등 6가지의 기하형태로 정의하여 구분하는 것이 편리하다. 분기점도 기하형태의 한 종류로 취급될 수 있겠으나 앞의 6가지 기하형태와는 정의방식이 다르기 때문에 독립적으로 취급한다.

2.1.3 정리 3

정점은 모든 이웃 절점들에 대해 UT_i 를 가지며

Table 1 Types of geometrical form and features

Type of geometric form	Topology of directed tree	Pieces of edges		Other features
Sink		UT _o	1	Any DT _i 's set are not one-sided for any half planes bisected by arbitrary DT _i
		UT _i	0	
		DT _o	0	
		DT _i	≥3	
Valley		UT _o	1	At least, 2 DT _i 's exist
		UT _i	1	
		DT _o	1	
		DT _i	≥2	
Ridge		UT _o	1	At least, 2 UT _i 's exist both side respect to the UT _o line
		UT _i	≥2	
		DT _o	1	
		DT _i	1	
Combine		UT _o	1	Only 2 UT _i 's exist in the one side respect to the UT _o line
		UT _i	2	
		DT _o	-	
		DT _i	-	
Apex		UT _o	0	At least 3 UT _i 's must exist
		UT _i	≥3	
		DT _o	1	
		DT _i	0	
Divide point		UT _o	≥2	Always occur after the VUT _o is applied
		UT _i	1	
		DT _o	-	
		DT _i	-	
Normal		UT _o	1	General case which is not applied to the above geometric form
		UT _i	1	
		DT _o	1	
		DT _i	1	

합물점은 모든 이웃 절점들에 대해 DT_i를 갖는다. 최대경사 상향 트리구조에 있어서 합물점은 항상 적어도 한 가지의, 연속적으로 연결된 UT_o들의 상향 경로를 통해 정점 또는 UT_o를 갖지 않는 경계점(boundary vertex)에 도달될 수 있다. 또한 UT_i를 갖지 않는 경계점도 항상 적어도 한 가지의, 연속적으로 연결된 UT_o들의 상향 경로를 통해 정점 또는 UT_o를 갖지 않는 경계점에 도달될 수 있다.

이같은 성질에 근거하여 볼때 합물점과 UT_i를

갖지 않는 경계점들은 최대경사 상향 트리의 경로를 결정할 때에 경로 출발점(path start point)들로 지정될 수 있다.

2.1.4 정리 4

유한의 삼각형요소화된 메쉬상의 모든 절점들은 UT 및 VUT들의 연속적인 연결로써 구성되는 최대경사 상향 트리에 반드시 참여한다. 그리고 경로 출발점으로부터 정점(또는 UT_o를 갖지 않는 경계점)에 이르는 모든 경로를 연결하면 메쉬상의 어떠한

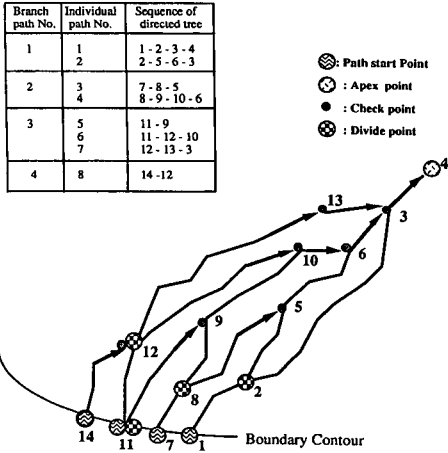


Fig. 3 Rule to select right path based on check points and return points

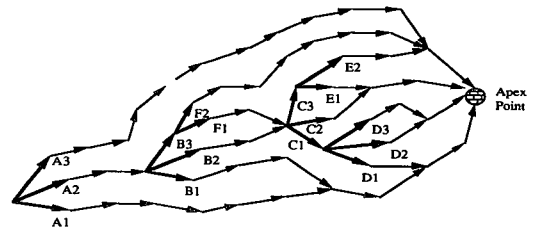
한 절점도 놓치지 않고 최대경사 상향 경로(steepest upward linked path)를 구성할 수 있다.

정리 4에 따른 최대경사 상향 경로에는 일반적으로 많은 분기된 경로가 존재한다. 곡면을 오른쪽부터 순차적으로 절삭가공하는 절차를 기준으로 하면, 이들 경로는 Fig. 3의 예와 같이 여러 갈래의 출발 경로와 분기된 경로가 있을 경우 오른쪽 경로가 왼쪽 경로보다 우선적으로 선택되는 우측 선행 규칙(rule to select right way)를 설정할 수가 있다. 그리고 이 규칙의 적용을 위해서는 정지점(check point)과 복귀점(return point)의 개념이 필요하다. 정지점은 한 갈래의 절삭경로가 종료되는 끝점으로 정의되고, 어느 절삭경로가 진행 중 정지점에 도달하면 절삭이송을 멈추고 급속이송으로 차기 절삭경로의 출발점으로 귀환하여야 하는데, 이 점을 복귀점으로 정의한다.

2.1.5 정리 5

최대경사 상향 경로들을 구성하는 데에 있어서 경로 출발점과 분기점은 복귀점의 역할을 담당하고, 정점은 항상 정지점의 역할을 담당한다. 합치점과 능선점은 적어도 한번의 절삭경로가 이 점들을 통과한 다음에 정지점의 역할을 담당한다.

일련의 UT_0 가 우측 선행규칙에 따라 연속적으로 연결될때 복귀점(경로 출발점 또는 분기점)을 출발하여 정지점에 도달할 때까지의 UT_0 들이 연결된 1개의 경로는 개별경로(individual path)라 정의하며, 경로 출발점에서 시작하여 다음의 경로 출



Path order	1	2	2a	3	3a	3b	4	5	6	7	7a	8	9	9a	10	11
Selected Return point	A1	A2	B1	B2	C1	D1	D2	D3	C2	C3	E1	E2	E3	F1	F2	A3
No. of point to return	2	1	3	2	4	6	5	4	3	2	3	2	1	2	1	0
Stored return points list	A2 A3		B2 B3 A3	B3 A3	C2 C3 B3 A3	D2 D3 C3 B3 A3	D2 D3 C3 B3 A3	C2 C3 B3 A3	C3 B3 A3	B3 A3	E2 B3 A3	A3	A3	F2 A3	A3	

Fig. 4 Dynamic storage and access of return points for a branch path

발점에 도달할 때까지 연결된 개별경로의 집합은 분단경로(branch path)라고 정의한다.

Fig. 3의 예를 통해 살펴보면 4개의 절점 1, 7, 11, 14는 경로 출발점의 역할을 담당하고 있으며, 결과적으로 4가지의 분단경로가 형성되고 있다. 경로 출발점 1에서 시작되는 첫번째 분단경로는 2개의 개별경로로 구성되는데, 첫번째 개별경로는 절점 1에서 시작하여 오른쪽 경로의 절점 2와 3을 연결한 다음 절점 4에서 정지(check)되고 있고 두번째 개별경로는 복귀점 2에서 시작하여 절점 5와 6을 지나 정지점 3에서 정지되고 있다. 첫번째의 개별경로가 진행되는 동안 분기점 2는 복귀점으로 지정되고 있고 합치점 3은 다음 개별경로들을 위한 정지점 후보(candidate)로 지정되고 있으며, 또한 두번째 경로가 진행되는 동안에도 합치점 5와 6은 정지점 후보로 지정되고 있다.

이같은 과정은 차기의 분단경로들을 형성하는 데에 있어서도 동일한 방법으로 적용되는데, 이때 개별경로들을 우측 선행규칙에 따라 연속적으로 결정하기 위해서는 동적인 복귀점의 저장 및 생성이 필요하고 이는 Fig. 4의 예에서 보이는 바와 같은 동적 복귀점 리스트(dynamic return points list)를 이용하여 실현될 수 있다.

Fig. 4의 예는 하나의 분단경로가 진행되는 동안 변화하는 동적 복귀점 리스트와 차기의 개별경로를 위해 선택되는 복귀점을 나타내고 있다. 이 예에서 처음의 경로 출발점은 분기점의 역할을 갖고 있기 때문에 복귀점으로 취급되며 가장 오른쪽의 루트

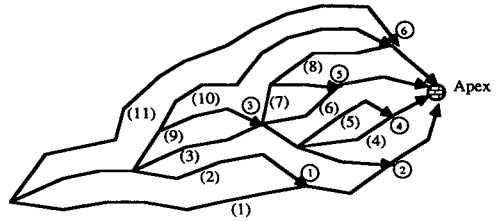
(route) A1은 현행 경로의 루트로 선택되고 나머지 루트 A2 및 A3는 순서대로 동적 복귀점 리스트에 저장된다. 현행 경로가 정지점을 만나서 1개의 개별경로가 완성되면 동적 복귀점 리스트의 맨 앞에 저장되어 있는 루트(A2)의 해당 복귀점으로 복귀한 다음 A2 루트를 따라 차기 개별경로가 진행된다, 이때 어느 한 루트가 차기의 경로 루트로 선택되면 이 루트는 자동적으로 동적 복귀점 리스트에서 제거된다. 한편 두번째 개별경로가 진행되는 가운데에도 다른 분기점 B를 만나면 여러 갈래의 루트 중 가장 오른쪽인 B1이 선택되고 나머지 B2 및 B3는 동적 복귀점 리스트에 남아 있는 루트들의 맨 앞부분에 차례대로 삽입되어 저장된다. 이와 같은 과정은 동적 복귀점 리스트에 저장된 루트들이 모두 소모될 때까지, 즉 하나의 분단경로를 구성하는 개별경로들 중 마지막의 개별경로가 선택될 때까지 진행됨으로써 하나의 분단경로를 완성하게 된다.

한편 Fig. 5는 각각의 개별경로가 진행되는 동안 어떻게 정지점들이 증가적으로 지정되고 저장되는 가를 보여 주고 있다.

2.2 최대경사방향 트리의 수치해석 과정

최대경사방향 트리는 삼각형요소화된 곡면메쉬에 대해 각 절점에 대한 상향 및 하향의 최대경사 에지(edge)들을 찾아 UT_0 및 DT_0 를 저장함으로써 시작된다. 그 후 각 절점에 대한 UT_i 및 DT_i 는 삼각형요소 이웃관계 리스트(adjacency list)의 상대관계(opposite relation)⁽¹⁸⁾를 적용하여 쉽게 결정된다. 이때 UT_i 및 DT_i 를 저장하는 데에 있어서는 정리 1에서 고찰된 바와 같이 하나의 절점에서 여러개 존재할 수 있기 때문에 존재 개수, 포인터(pointer) 그리고 요소 포인터(element pointer)로 구분하여 저장하였다. Fig. 6은 제시된 유한 메쉬상의 각 절점에 있어서 UT_0 와 UT_i 들의 저장되는 구조를 예시한 것이다.

초기의 UT 및 DT 들에 관한 정보들이 완성된 다음에는 Table 1에서 제시된 바와 같은 기하형태들의 분류정보들이 생성되는데, 이는 Fig. 7의 흐름도에서 보이는 바와 같이 각 절점들에 있어서의 UT_i , UT_0 , DT_i , DT_0 들의 존재성과 존재개수 그리고 상대 방향각 등을 이용하여 쉽게 인식할 수 있도록 하였다. 이는 정리 3과 Table 1에서 제시된 바와 같은 토폴로지를 이용한 것인데, UT_0 가



Check point number	Individual path index number										
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
①	○	●	○	○	○	○	○	○	○	○	○
②	○	○	●	○	○	○	○	○	○	○	○
③			○	○	○	○	○	○	○	●	○
④				○	●	○	○	○	○	○	○
⑤						○	●	○	○	○	○
⑥								○	○	●	○
Apex	●	○	○	○	○	○	○	○	○	○	○

○ : Designated to a Check point
● : Effective Check point to stop an Individual path

Fig. 5 Designation of check points from the previous path history to stop each individual path

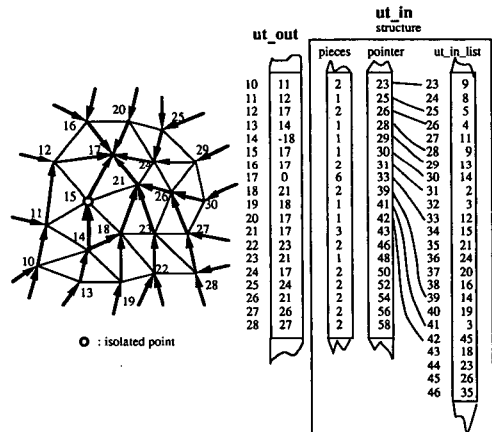


Fig. 6 Data structure example in the case of upward directed tree

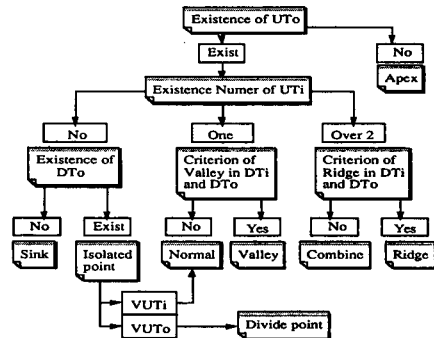


Fig. 7 Process for the recognition of geometric form types

존재하지 않는 것은 유일하게 정점이기 때문에 제 1단계에서는 이를 결정하고 다음 단계에서는 UT_i 의 존재 개수를 기준으로 3가지의 그룹, 즉 합물점과 고립점(UT_i 의 개수=0) 그룹, 보통점과 계곡점(UT_i 의 개수=1) 그룹, 그리고 합치점과 능선점(UT_i 의 개수 ≥ 2) 그룹으로 분류되게 하였다. 제3단계에서는 각 그룹에 대해 DT_0 의 존재유무, 방위각 및 에지의 상하 테스트를 통한 계곡점 및 능선점의 추출 기준을 이용하여 6가지의 기하형태가 지정되도록 하였다. 한편 이때 고립점은 제4단계 과정에서 VUT_i 의 상대관계에 의해 연결된 후 보통점으로 변환되도록 하였고 이 점의 상대점에 분기점으로 지정(UT_0 의 요소를 음으로 표현)함과 동시에 위의 6가지 기하형태와는 독립적으로 마련된 분기점 리스트에 생성되거나 추가되도록 하였다.

다음으로는 정지점 후보들을 저장하는 과정이 실시되는데, 이는 UT_i 의 개수가 2개 이상인 절점(합치점 및 능선점)들을 위해 정지점 리스트에 지정하는 것으로 하였다. 또한 경로 출발점 리스트에 있어서는 합물점을 찾아서 낮은 순서부터 먼저 저장하고, 그 다음으로 UT_i 를 갖지 않는 경계점들을 메쉬의 경계를 따라 시계방향으로 따라 탐색하면서 순서대로 저장되도록 함으로써 우측 선행규칙의 적용이 용이하도록 하였다.

이와 같이 정지점 및 경로 출발점 리스트의 저장이 완료된 이후에는 각각의 경로 출발점에서 시작하여 UT_0 를 연속적으로 연결함으로써 최대경사방향 트리가 구성된다. 앞의 Fig. 4에서 예시되었던 바와 같이 현행점이 분기점이 아닐 경우에는 단순히 현행점의 UT_0 에 연결하고, 분기점일 경우에는 우측 선행 규칙에 따라 가장 오른쪽의 UT_0 가 선택된 후 나머지 UT_0 들은 동적 복귀점 리스트의 저장요소들을 변경시키는 데에 사용된다. 그리고 이같은 과정이 진행되다가 현행점이 정지점으로 인식되는 경우에는 1개의 개별경로를 완성하고 복귀점 리스트의 첫번째 저장요소를 차기 개별경로의 출발점으로 선택하여 위의 과정이 다시 실행된다. 한편 이들 과정중 1개 분단경로의 마지막 개별경로에 해당하는 경로에서는 현행점이 정지점에 도달할 때 복귀점 리스트의 저장요소가 비워지기 때문에 저장요소의 개수가 0일 때에는 1개의 개별경로 및 분단경로를 완성함과 동시에 경로 출발점 리스트의 다음 저장요소를 취하여 차기 분단경로의 연속적인

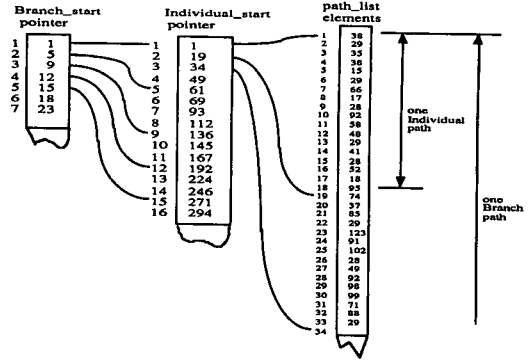


Fig. 8 An illustration of data structure for developing the individual and branch path list

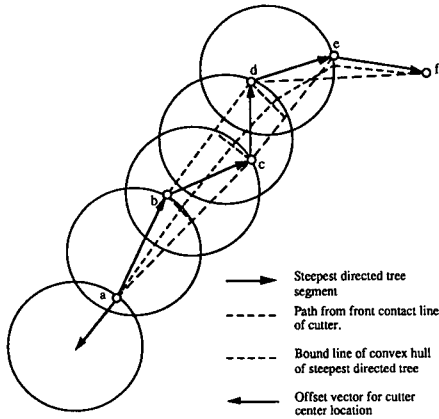
링크를 실행한다. 이와 같은 과정은 등지(nest)형 논리를 통하여 쉽게 처리될 수 있으며, 앞의 정리 4에 의거하여 유한한 삼각형요소화된 메쉬상의 어떠한 절점도 놓치지 않고 최대경사 상향 경로에 참여하도록 구성할 수 있다.

Fig. 8은 이들 실행과정에서 순차적으로 생성되는 개별 및 분단경로 리스트의 데이터 구조를 예시한 것이다.

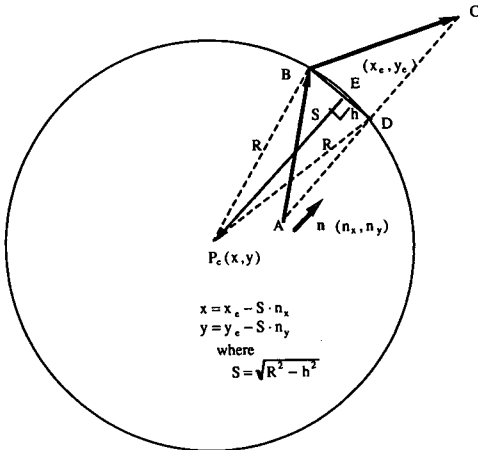
2.3 xy-평면상의 커터 중심위치의 결정

최대경사 상향 트리가 엔드밀 커터와 곡면모델과의 접촉점들로서 직접 사용되는 데에는 다소의 문제점이 생긴다. 일반적으로 삼각형요소화된 곡면에서 구해진 최대경사 상향 트리의 에지들은 지그재그 모양으로 연결되는 부분이 많기 때문에 이로 인하여 커터 중심의 경로가 좌우방향으로 요동하게 되며 절삭성과 표면 품질이 열악해 짐은 물론 곡면과 커터 사이의 간섭 확률이 더욱 높아진다. 그리고 최대경사 상향 트리의 에지들을 Bezier 또는 B-spline으로 보간하는 경우에도 진동 현상을 방지하기는 어렵다. 따라서 본 연구에서는 곡면의 이상적 최대경사 방향선이 최대경사 상향 트리의 폴리라인(polyline)에 대한 콘벡스 헐(convex hull)내에 존재한다는 가정하에서 각 개별경로마다 최대경사 상향 경로의 절점들을 그대로 접촉하게 하면서 커터 중심의 운동을 de Casteljau 알고리즘⁽¹⁹⁾에서의 분할원리(degree 3의 polynomial curve)를 이용해 근사 평활화하는 방법을 마련하였다.

Fig. 9(a)를 예로 들어보면 절점 A로부터 f까지의 개별경로가 있을때 각각의 2개 에지 짝에 대한 2등분선(Fig. 9(b)의 선분 BD)들을 구하여 이들의



(a) Locus for positioning in xy-plane from the leading trail tips of cutter



(b) Determination of cutter center location

Fig. 9 Adjustment of trailing edges for an individual path and determination of cutter center location

수직방향으로 커터의 중심점이 위치되도록 하였고, 커터 운동방향의 선단원호는 현행 절점(Fig. 9(b)의 B)와 상대교점(Fig. 9(b)의 D)를 지나도록 하였다. Fig. 9(b)는 2개 예지 짝에 대한 커터의 오프셋 방향과 중심위치의 산출원리를 보여주고 있다. 한편 개별경로의 첫 절점에 있어서는 차기 절점에 대한 커터의 오프셋방향과 동일하게 취하였으며 끝 절점에 있어서는 적전 절점에 대한 오프셋방향과 동일하게 취하였다. 그렇게 함으로써 커터의 운동 방향에 대한 선단점의 이동경로는 최대경사 상향 트리로 구성된 폴리라인의 콘벡스 헐내에서 좌우의

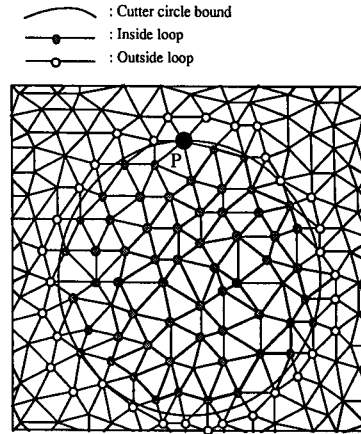


Fig. 10 Configuration of interference checking in cutter circle bound

진동현상없이 움직일 수 있도록 하였다. 한편 개별 경로에 대한 시작운동이 직전 개별경로들과 접속되도록 하기 위해서 현행 개별경로의 출발점(복귀점)을 통과했던 최후의 커터위치를 탐색하여 이를 직전 개별경로가 완료된 후에 복귀되는 커터 중심위치로 하였고, 이를 현행 개별경로에 대한 초기의 커터위치 앞에 삽입하였다. 또한 경로 출발점이 함물점의 기하형태를 띠게 될 때에는 커터 중심위치가 중복되는 것을 피하기 위해서 커터중심이 함물점에 위치하는 커터크기의 원 내부에 있을 때에는 생략되도록 하였다.

2.4 입형 간섭체크에 의한 커터높이 결정

평면상에서 평활화된 커터의 중심 위치들이 결정된 다음에는 커터가 삼각형요소화된 곡면과의 간섭이 일어나지 않도록 커터의 높이값(z value)들을 결정하였다. 간섭체크 모델로서는 xy 평면상의 삼각형요소화된 그물에 커터 원호가 매핑되게 하는 모양을 취하고 이 영역에서 높이방향으로 간섭이 생기지 않는 가장 낮은 위치를 결정하는 방법을 사용하였으며 이는 Fig. 10에서 보이는 바와 같다.

첫단계에서는 예상 접촉점 P(현행 절점)로부터 커터 원호를 둘러싸는 내부 루프(inside loop)와 외부 루프(outside loop)를 구하는데, 이는 커터 원호에 대한 최근접 내부 및 외부 경로(nearest inside and outside path)를 따라서 Jarvis march⁽²⁰⁾를 반복함으로써 탐색되었다. 그리고 이를 진행하는 과정에 있어서는 동시에 커터 원호와 교차하는 횡단

에지들의 양단 절점 리스트들도 함께 결정되도록 하였다.

두번째 단계에서는 커터 원호 내부에 존재하는 모든 절점들을 결정하였는데 이는 삼각형요소화된 메쉬의 이웃관계 성질을 내부 루프에 적용하는 shrinking loop법⁽²⁰⁾을 이용하여 결정되도록 하였다. 그리고는 이들 절점들 중 높이값이 가장 큰 것을 취하여 최고 내부 높이로 설정하였고, 또한 외부 루프를 구성하는 절점들 중 가장 높은 점을 취하여 이를 최고 외부 높이로 설정하였다.

한편 세번째 단계에서는 최고 내부높이와 최고 외부높이를 비교하여 최고 내부높이가 최고 외부높이보다 클 경우에는 커터의 간섭이 내부에서 일어나는 것으로 인정하게 이 최고 내부높이값을 커터의 결과 높이값으로 결정하였다. 그러나 이의 역관계에 해당되는 경우에는 커터와 곡면과의 접촉점이 커터의 원호상에 존재할 수 있는 것으로 인정하였다.

네번째 단계에서는 이 접촉점을 커터 원호와 횡단 에지들과의 교차점들로부터 결정하는 과정이 실행되었는데, 이때 횡단 에지의 양단 절점의 높이값이 모두 최고 내부높이값보다 작을 경우에는 교차점도 최고 내부높이값보다 작기 때문에 교차점을 구하는 백터연산은 오직 횡단 에지의 양단 절점 높이들 중 어느 하나가 최고 내부 높이 값보다 클 경우에만 적용되도록 하였다. 따라서 이같이 추출된 횡단 에지들에 대해 커터 원호와의 교차점들이 계산되고 이들 교차점들 중 가장 큰 값을 취한 다음 이를 다시 최고 내부 높이값과 비교하여 그 큰 값을 결과적인 커터의 높이값으로 결정하였다.

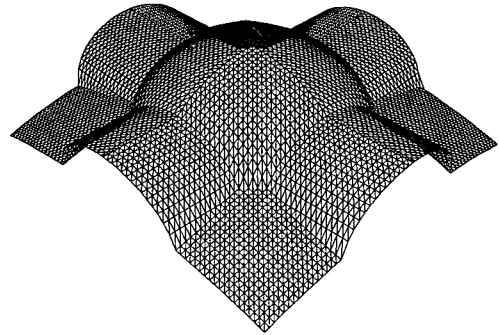
마지막 단계에서는 결정된 커터 높이값과 예상 접촉점의 높이값(공차 포함)을 비교하여 커터 높이값이 클 경우에는 현행 절점에서 간섭 현상이 생기는 것으로 취급하고 이를 간섭 리스트의 현행 절점에 해당하는 레코드에 높이 차이값과 함께 기록되도록 하였다.

3. 적용성능의 평가

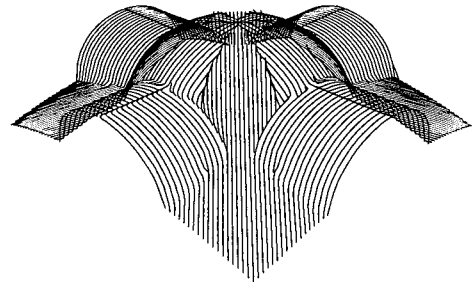
3.1 가공성능의 평가 및 고찰

최대경사방향 트리추적법의 적용성능과 장점, 그리고 한계성 등을 조사하기 위해서 다음 3종류의 원형 모델을 대상으로 가공실험을 실시하였다.

첫번째 모델은 Fig. 11에서 보이는 바와 같이 2



(a) Triangulated surface model



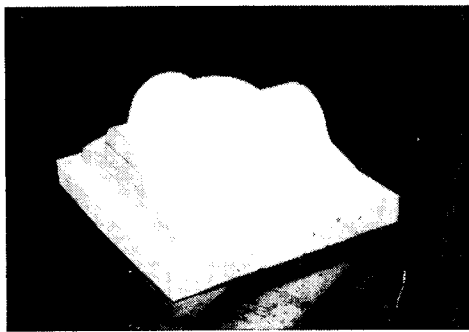
(b) Cutter path expressed by cutter contact points

Fig. 11 Prototype used for the first machining test

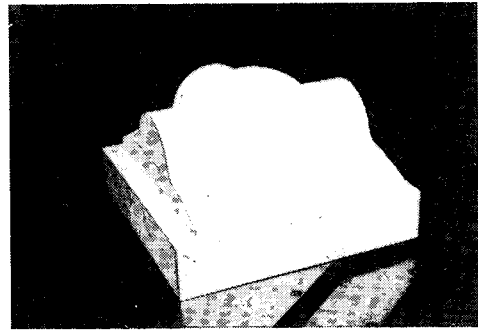
개의 원통면과 2개의 구면으로 형성된 복합곡면으로 모델크기는 132×144×92 mm이며 삼각형요소화 과정에서의 grid 크기는 2.25 mm이다. 입력점에 대해 삼각형요소화된 곡면형상은 Fig. 11(a)와 같으며, 최대경사방향 트리추적법을 적용하여 얻어진 커터경로(커터와 곡면의 접촉점들로 표현)는 Fig. 11(b)에서 보이는 바와 같다. 한편 가공실험을 실시하는 데에 있어서는 Table 2에서 제시된 바와 같이 4가지의 실험조건을 사용하였는데, Case A와 Case B는 각각 9.525 mm(3/8 inch)와 12.7 mm(1/2 inch) 직경의 엔드밀 커터를 사용하면서 최대경사방향 트리추적법이 적용된 경우이고, Case C와 Case D는 각각 12.7 mm 직경의 볼밀과 12.7 mm의 엔드밀 커터를 사용하면서 CC-Cartesian 방법(Albacore사의 GSURF 시스템을 이용)이 적용된 경우이며, CC-Cartesian 방법이 적용된 경우 cross feed의 크기는 삼각형요소화 모델의 grid 크기와 같이 2.25 mm로 하였다. 결과적으로 절삭된 4가지의 가공부품의 모습은 Fig. 12(a)~(d)에서 보이는 바와 같다.

Table 2 Specification for the machining test environments

Kinds of test conditions	Case A	Case B	Case C	Case D
Cutter path planning method	Steepest directed tree method	Steepest directed tree method	CC-Cartesian method	CC-Cartesian method
Cutter type	End mill	End mill	Ball mill	End mill
Cutter diameter	9.525 mm (3/8 inch)	12.7 mm (1/2 inch)	12.7 mm (1/2 inch)	12.7 mm (1/2 inch)



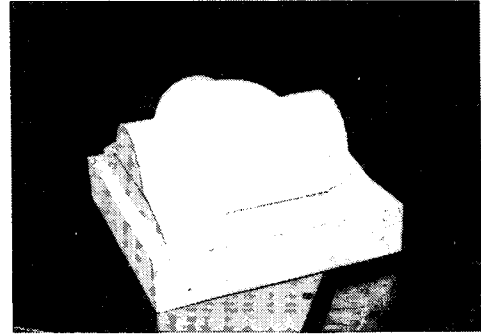
(a) Case A



(c) Case C



(b) Case B



(d) Case D

Fig. 12 Photo of resultant machined parts in each test condition

한편 이들을 평가하기 위한 기준으로는 Table 3에 제시된 바와 같이 최대커슝높이, 비절삭 체적, 동적 절삭상태, 가공시간 등의 4가지로 하였다. 여기서 최대커슝높이는 4부분의 원통 상면과 4부분의 구면 위에 40 mm 길이의 거칠기 곡선을 레이저 스캐너로 채취한 다음 8가지 최대높이 거칠기값 중에서 가장 큰 값을 선택하는 것으로 하였다. 그리고 비절삭 체적의 기준으로는 원통면과 구면들 사이의 12가지 교선부위에 남아있는 비절삭 체적부분을 원

호게이지를 이용하여 원호 반경값을 근사적으로 구하고 그 중 최대값을 취하였다. 동적 절삭상태에 있어서는 가공중 플런지컷 패턴이 생기는 부위(CC-Cartesian 방법에서 절삭이송방향과 cross feed 방향 모두가 아래 방향인 부분으로 원통면 뒷쪽의 좌측부위)를 절삭할 때 생기는 절삭음(cutting acoustics)을 정성적으로 비교하였고, 이때를 기준으로 절삭가능한 최고 이송속도를 NC제어반의 dryrun 다이얼을 이용하여 근사적으로 설정하였

Table 3 Comparison of resultant machining performance indices for different test conditions

Performance indices for comparison	Case A	Case B	Case C	Case D
Maximum cusp height	0.085 mm	0.040 mm	0.275 mm	1.430 mm
Maximum radius of uncut volume	2.0 mm	3.4 mm	6.5 mm	4.0 mm
Dynamic cutting state(qualitative)	stable	very stable	sometimes unstable	sometimes unstable
Total machining time(expected) [Possible feedrate]	28.9 min. [600mm/min.]	25.6 min. [700mm/min.]	35.2 min. [400mm/min.]	45.8 min. [300mm/min.]

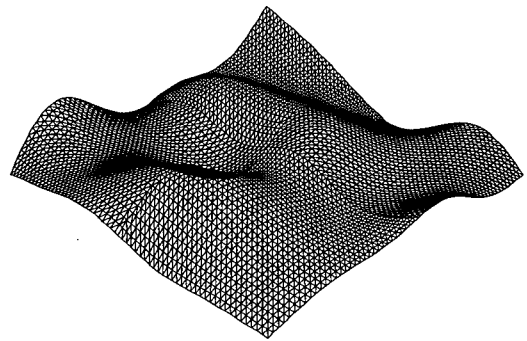
다. 또한 총 절삭시간은 이때 설정된 최고 가능 이송속도로 가공될 경우를 기준으로 하여 총 이송거리를 이송속도로 나누고 여기에 급속이송시간을 추가시켜 결정하였다.

이때 얻어진 결과들은 Table 3에 제시되었는데, 같은 크기의 엔드밀과 볼밀이 각각 최대경사방향 트리추적법과 CC-Cartesian 방법에 대해 사용된 Case B와 Case C를 비교하여 보면 표면거칠기는 6.8배, 비절삭 체적의 최대반경은 1.8배 정도로 Case B의 경우가 월등히 좋아지고 있고, 동적 절삭상태도 Case B의 경우가 많이 좋아지고 있음은 물론 설정 가능한 최대 이송속도로 가공할 때의 절삭시간도 Case B의 경우가 27% 가량 절감되고 있음을 보이고 있다.

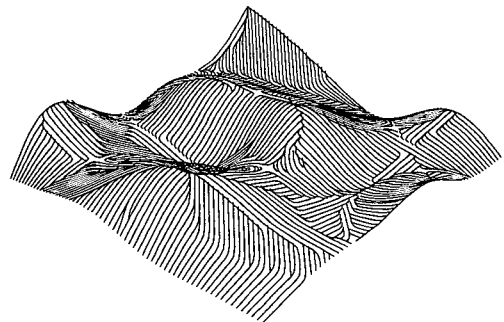
같은 크기의 엔드밀들이 각각 최대경사방향 트리추적법과 CC-Cartesian 방법에 사용된 Case B와 Case D를 비교하여 보면 표면거칠기는 35.7배, 비절삭 체적의 최대반경은 1.2배로 Case B의 경우가 월등히 좋아지고 있고, 동적 절삭상태도 아주 좋아지고 있을 뿐 아니라 절삭시간도 40% 정도 단축되고 있어서 최대경사방향 트리추적법의 적용효과가 매우 두드러지게 나타나고 있음을 보여 주고 있다.

한편 작은 직경의 엔드밀이 사용된 Case A를 Case B와 비교해 보면 표면거칠기는 다소 나빠지고 있으나 비절삭 체적을 적게 남기는 데에는 오히려 더 효과적임을 알려주고 있으며, 이는 Case C와 비교하여 볼 때에도 모든 평가 기준에 대해 향상되고 있음을 보여 주고 있다. 그리고 이 복합곡면 모델에서는 CC-Cartesian 방법이 적용될 때 커

터 경로가 원통면과 구면과의 교선에 평행하게 이동하는 도중 약간의 과절삭이 발견되고 있으나 최대경사방향 트리추적법이 적용된 경우에는 어느 부분에도 과절삭이 발견되고 있지 않았다. 이는 최대



(a) Triangulated surface model



(b) Cutter path expressed by cutter contact points

Fig. 13 Prototype model used in the second machining test

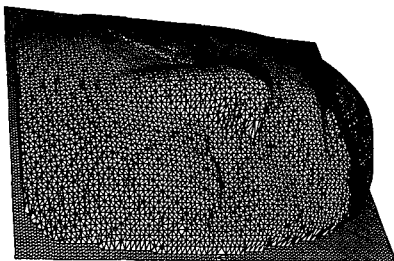
경사방향 트리추적법이 불연속면이나 교선 부위에 있어서도 간섭 체크가 완벽하게 이루어 지고 있으며 효과적으로 적용될 수 있음을 검증해 주는 것이라 하겠다.

두번째 모델은 Fig. 13에서 보이는 바와 같이 지형(geographical form) 모양의 순수 자유곡면 모델인데 모델크기는 $122 \times 122 \times 65$ mm이며 grid 크기는 2.0 mm이다. 사용된 엔드밀의 직경은 9.525 mm였으며 이 모델에서는 앞의 Table 1에서 정의되었던 기하형태들이 모두 포함되어 있다. 내부에 2개 포함이 되어 있는 함몰점에 있어서는 미리 드릴가공을 이용하여 충분히 큰 안내구멍을 가공하여 놓았다. 삼각형요소화된 곡면형상은 Fig. 13(a)에서 보이는 바와 같으며, 최대경사방향 트리추적법을 적용하여 얻어진 커터 경로는 Fig. 13(b)에서 보이는 바와 같다.

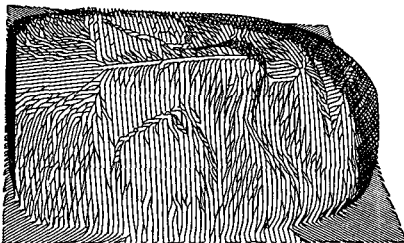
전체적으로 볼때 함몰점에 있어서의 원호 마크를 제외하면 가공면의 표면거칠기는 앞의 첫번째 모델보다도 더욱 매끈하게 가공되고 있었으며, 모든 경우의 기하형태들에 대하여 예외없이 잘 적용되고 있음을 고찰할 수 있었다. 한편 본 연구의 경우에는 간섭 체크를 실행하는 과정에서 커터 간섭에 의한 비절삭 부위가 생기면 비절삭 부위의 두께와 함께 이를 간섭 리스트 화일에 기록하고 있어서 작은

크기의 엔드밀로 교체한 후 이 부분을 추가 가공하면 함몰점 부위의 비절삭 체적을 더욱 적게 남기고 가공할 수 있다.

세번째 모델은 Fig. 14에서 보이는 바와 같이 마스크(indian mask)형 조각모델인데, 복잡한 형상의 날카로운 에지들을 많이 갖는 조각형 곡면에 대한 본 알고리즘의 적용성능을 평가하기 위한 것이다. 이 모델의 3D 입력데이터는 가로 및 세로방향에 대해 레이저 스캐닝(laser scanning)된 데이터들을 구간 보간법에 의해 여과시킨 다음 이를 중첩시켜 사용하였으며,⁽²⁾ 모델크기는 $188 \times 152 \times 105$ mm, grid의 평균 크기는 1.2 mm이고 총 절점의 수는 5,890개이다. 이 모델에 있어서도 함몰점이 내부에 3개(입 속에 1개, 눈과 코 사이에 2개) 포함되어 있어서 이 절점들에는 미리 충분히 큰 안내구멍을 가공하여 놓았고, 엔드밀은 9.525 mm 직경의 것을 사용하였다. 삼각형요소화된 곡면형상은 Fig. 14(a)에서 보이는 바와 같으며, 최대경사방향 트리추적법을 적용하여 얻어진 커터 경로는 Fig. 14(b)에서 보이는 바와 같다.



(a) Triangulated surface model



(b) Cutter path expressed by cutter contact points

Fig. 14 Mask model used in the third machining test



(a) 2nd sculptured surface model



(b) 3rd carved mask model

Fig. 15 Photos of machined parts for the second and third model

본 모델의 경우에는 곡면이 복잡하고 입력 데이터들도 불규칙하게 배열되는 경우가 많아서 절삭경로도 다소 불규칙하고 복잡하게 나타나고 있다. 절삭된 가공물의 모양은 앞의 두번째 모델과 함께 Fig. 15에서 보이고 있다. 가공면을 살펴볼 때 곡률이 작은 면들에 있어서는 대체로 가공면이 양호하나 약간의 격자형 커슴들이 남아 있는데 이는 입력점들의 불규칙성에서 비롯되는 것으로 판단된다. 더우기 마스크 외곽의 경계부위와 눈주의, 그리고 콧날 아래부위들에 있어서는 가공면이 심하게 울퉁불퉁한데, 이는 레이저 스캔된 데이터의 특성상 경사도가 큰 측면들에서 입력데이터의 밀도가 낮게 되고 이로 인해 fitting 과정에서의 오차를 수반하기 때문에 발생된 것이라 판단된다. 그리고 이들 부위들에서 과절삭이 일어나고 있는지를 점검하기 위해 실물 마스크와 가공된 모델을 비교하여 보았는데 대부분 돌출된 비절삭 체적이었으며 과절삭이 일어난 부분은 좀체로 찾을 수가 없었다. 이들 돌출부위들에서는 데이터들의 불규칙성으로 인해 경사도가 큰 경계부위에서 부드럽지 못한 삼각형요소들이 발견되고 있었고 이로 인해 절삭경로의 산출시 간섭 체크과정에서 비절삭 체적들이 많이 남음으로서 생성된 것으로 생각된다. 다음으로 날카로운 에지부분의 절삭되는 모양을 점검하여 보면, 눈섭부분과 같이 볼록한 에지부분은 잘 가공이 되고

있었으나 오목한 부분의 에지들은 비절삭 체적으로 남아 있었으며 몇몇 불완전한 부분들에 있어서는 날카로운 에지가 둔화되는 곳도 발견되고 있었다. 따라서 향후 rapid prototyping을 위한 가공방법으로 본 알고리즘을 실용적으로 적용하기 위해서는 날카로운 에지들을 정확하게 표현할 수 있는 레이저 스캐닝방법과 형상 모델링방법, 그리고 커터의 직경이 큰 것으로 1차 가공한 후 비절삭 부위를 작은 커터로 2차, 3차 가공하는 다단계 가공에 대한 연구가 계속 진행되어야 할 것이다.

3.2 알고리즘의 처리속도 및 고찰

본 연구에서 추구하는 접근방법들이 실시간 이내에서 처리속도상 실용성이 있는가를 점검하기 위해 앞의 가공실험에서 사용하였던 첫번째 모델(복합곡면형)을 대상으로 알고리즘들의 각 단계별 계산시간을 체크하였다. 본 알고리즘들은 ANSI-C 언어로 작성되었고 SUN SPARC-I workstation (25 MHz)의 UNIX 환경하에서 C 컴파일러를 이용하여 처리되도록 하였다. 처리단계로는 Table 4에서 보이는 바와 같이 2가지의 그룹(곡면모델링 그룹과 절삭경로 산출그룹)으로 나누었으며 곡면모델링 그룹에서는 3개의 과정, 절삭경로 산출그룹에서는 5개의 과정으로 나누어 체크하였다. 그리고 실행시간은 UNIX 환경에서 Clock 함수를 이용하여 milisecond

Table 4 Comparison of CPU running time in each test

Group	Procedures	N	Running time(sec.)			
			1000	2000	4000	8000
Triangulated surface model	Main triangulation		3.566	8.752	31.593	78.663
	Mesh optimization		14.173	33.147	84.618	186.159
	Subdivision and decimation		2.455	6.216	14.148	33.459
	Local sum		20.194	48.115	130.359	298.281
Cutter path planning	Tree edge information		0.421	0.698	1.549	3.283
	Geometric form type		0.150	0.268	0.536	0.983
	Tree sequencing		0.032	0.049	0.092	0.149
	Planetary positioning		1.260	2.452	4.779	8.978
	Buit-in interference check		2.838	7.536	18.803	48.715
	Local sum		4.701	11.003	25.759	62.112
	Total sum		24.895	59.118	156.118	360.393

단위로 산출되었는데, 타이머에서의 불규칙 오차들을 제거하기 위해서 5회 반복하여 구한 다음 이들의 평균값을 취하였다. 적용모델에 대한 입력점 개수 N 은 1000개로부터 8000개까지 2배씩 증가하는 구간을 취해 $C \cdot N^P$ 모델에서의 계수 P 값에 따라 입력점 개수의 증가에 따른 실행시간의 증가율을 점검할 수 있도록 하였다. 삼각형요소화 곡면모델링 그룹에서 subdivision 및 decimation 과정에서의 계산소요시간은 지정된 공차값의 크기에 의존하는데, 여기서 공차값은 메쉬 최적화과정에서 계산되는 아르펜차 높이들의 평균값을 구하여 이 값의 1/4 크기가 지정되도록 하였고, 이때 절점의 수는 5~8% 가량이 증가되었다.

결과적으로 얻어진 계산소요시간들은 Table 4에 제시된 바와 같다. 전반적으로 살펴보면 곡면모델링 그룹은 절삭경로 산출그룹보다 약 4배 정도로 계산시간이 많이 소요되고 있음을 보이고 있다. 그리고 각 과정중 가장 시간이 많이 소요되는 과정은 메쉬 최적화과정이었으며 그 다음은 메인 삼각형요소화 과정이었고, 입형 간섭체크과정이 3번째로 시간이 많이 소요되었다. 여기서, subdivision 및 decimation 과정의 계산시간은 상대적으로 적게 소요되었으나 이 과정은 삼각면의 공차값이 작게 지정될 경우 그리고 적용모델의 곡면형상이 복잡할수록 오랜 시간이 소요된다.

이들 소요시간들을 $C \cdot N^P$ 모델에 따라 계수 P 의 값을 산출해 보면 모두 1.3 이내에 들어오고 있다. 특히 입형 간섭체크과정에 대한 P 값은 약 1.21로 계산이 되었는데 이는 불밀을 사용하고 있는 상용의 간섭체크 알고리즘(이들의 경우 P 는 1.5 이상의 값을 갖는 것으로 보고⁽⁶⁾되고 있음)들과 비교해 볼때 월등히 작은 값이다. 이 효과는 본 연구의 입형 간섭체크 알고리즘이 반복성연산(iterative calculation)을 포함하지 않고 있고 오직 교차점 산출을 위한 벡터 연산과정만이 계산시간을 많이 요구하는 연산요소이며, 또한 커터 원호와 횡단 모서리와의 교차점 산출에 있어서도 모서리의 양단점이 모두 커터 원호의 내부 최고점보다 낮을 때에는 이 연산과정이 생략될 수 있는 특성을 가지고 있어서 커터 원호의 크기가 많이 커지거나 곡면모델의 절점수가 많아지더라도 교차점 산출과정이 실행될 경우는 별로 많아지지 않기 때문이다. 또한 이 특성을 확인할 목적으로 각각의 1회 간섭체크과정마다 교차점이 산출되는 경우의 수에 대한 평균

값을 산출하였는데(총 교차점 산출회수의 합/총 절점수), 본 적용모델의 경우는 2.67의 값으로 계산되었다. 따라서 본 간섭체크 알고리즘이 실행되는 동안 계산시간을 많이 요하는 연산요소들은 교차점 산출과 같은 대수연산과정보다는 Jarvis march 및 Shrinking loop법에 의한 해당 절점들의 탐색과정이었음이 확인되었고, 이 특성은 종래의 간섭체크 알고리즘이 계산 소요시간의 문제로 인해 곡면모델의 해상도를 높일 수 없었던 단점을 해결해 줄 수 있는 성질이라 하겠다.

한편 절삭경로 산출그룹에서 입형 간섭체크과정 다음으로 계산을 많이 필요로 하는 과정은 xy -평면 상의 커터 중심위치를 계산하는 과정이었으며, 이에 비해 최대경사방향 트리를 결정하는 과정이나 절점의 기하학적 형태를 인식하는 과정, 그리고 상향 트리를 링크하는 과정 등에서는 비교적 적은 계산시간이 소요되고 있음을 알 수 있었다. 그리고 절삭경로 산출그룹의 전체 계산소요시간은 삼각형요소화 곡면모델링과정의 것에 비해 약 20% 내외로 적게 소요되고 있었으며, 그 결과 본 연구의 최대경사방향 트리추적법은 계산소요시간의 측면에 있어서도 효과적인 방법임을 고찰할 수 있었다.

4. 결 론

본 연구에서는 삼각형요소로 표현된 곡면을 엔드밀 커터를 이용해 곡면의 최대 경사선을 따라상향으로 NC 절삭하게 하는 최대경사방향 트리추적법을 개발하였다. 이 방법은 삼각형요소화 과정을 통해 형상공차가 가급적 균일하게 유지되도록 곡면모델이 준비되면, 곡면 위의 각 절점에 대한 토폴로지들을 이용하여 최대경사방향 트리들을 연속적으로 추적하고, 이 최대 경사선의 평활화과정과 커터의 간섭체크과정을 거치면서 최종의 공구위치 산출데이터(CL data)를 얻어내고 있다.

5가지의 정리들과 6가지의 기하형태들의 정의에서 출발된 최대경사방향 트리의 결정 알고리즘은 수치화 과정에서 여러 형태의 곡면모델에 대해 예외의 경우가 없이 잘 적용되었으며, 상향 트리의 평활화 알고리즘에 있어서도 커터의 좌우 요동이나 cross feed의 크기 변화가 없이 커터를 부드럽게 잘 움직여 주었다. 그리고 입형 간섭체크 알고리즘도 연산의 안정성과 높은 해상도를 유지하면서 빠른 속도로 잘 처리되고 있음을 확인할 수 있었다.

또한 이 방법은 복합곡면, 자유곡면, 조각곡면 등의 모델을 대상으로 불밀을 사용하는 상용의 CC-Cartesian 방법과 가공성능이 비교되었는데 우수한 장점들을 나타내었다. 구면과 원통면으로 구성된 복합곡면의 절삭실험에서 가공면의 최대커슝 높이는 같은 크기의 불밀을 사용하는 CC-Cartesian 방법의 경우에 비해 6.8배나 좋아졌으며, 곡면의 오목한 교차 부위에 있어서의 비절삭 체적도 1.8배나 적게 남았다. 그리고 동적 절삭상태도 매우 안정하여서 가능한 최대 이송속도로 가공할 때의 절삭시간도 27% 정도 절감되고 있음을 보였다. 이와 같은 결과는 지형 자유곡면 모델의 절삭실험에서도 비슷한 장점을 보였으며, 날카로운 모서리 부위가 많이 포함된 마스크형 조각모델의 경우에는 더욱 효과적으로 과절삭없이 가공되고 있음이 검토되었다.

후 기

본 연구는 한국과학재단의 연구비(913-0909-006-1) 지원에 의해 이루어졌으며, 이에 관계자 여러분께 감사사를 드립니다.

참고문헌

- (1) Loney, G. D. and Ozsoy, T. M., "NC Machining of Free Form Surfaces," *Computer Aided Design*, Vol. 19, No. 2, pp. 85~90.
- (2) Bradley, C. and Vickers, G. W., 1992, "Automated Rapid Prototyping Utilizing Laser Scanning and Free-Form surface Machining," *Annals of the CIRP*, Vol. 41, No. 1, pp. 437~440.
- (3) Choi, B. K., Lee, C. S., Hwang, J. S. and Jun, C. S., 1988, "Compound Surface Modelling and Machining," *Computer Aided Design*, Vol. 21, No. 3, pp. 181~195.
- (4) Wang, H., Chang, T. C. and Wysk, R. A., 1987, "On the Efficiency of NC Tool Path Planning for Face Milling Operations," *ASME Journal of Engineering for Industry*, Feb., Vol. 111, pp. 22~26.
- (5) Hansen, A. and Arbab, F., 1988, "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.

- (6) Choi, B. K. and Jun, C. S., 1989, "Ball-end Cutter Interference Avoidance in NC Machining of Sculptured Surfaces," *Computer Aided Design*, Vol. 21, No. 6, pp. 371~378.
- (7) Dong, Z., Li, H. and Vickers, G. W., 1993, "Optimal Rough Machining of Sculptured Parts of a CNC Milling Machine," *ASME Journal of Engineering for Industry*, Nov., Vol. 115, pp. 424~431.
- (8) Catania, G., 1992, "A Computer-Aided Prototype System for NC Rough Milling of Free-Form Shaped Mechanical Part-Pieces," *Computers in Industry*, Vol. 21, pp. 275~293.
- (9) Suh, Y. S. and Lee, K., 1990, "NC Milling Tool Path Generation for Arbitrary Pockets Defined by Sculptured Surfaces," *Computer Aided Design*, Vol. 22, No. 5, pp. 273~284.
- (10) Guyder, M. K., 1990, "Automating the Optimization of 2.5 Axis Milling," *Computers in Industry*, Vol. 15, pp. 163~168.
- (11) Aoyame, H. and Kishinami, T., 1987, "Study on Development and Cutting Performance of Elliptic Ball End Mill," *Journal of Japan Society of Precision Engineering*, Vol. 53, No. 3, pp. 461~466.
- (12) Maeng, H. Y. and Park, C. K., 1990, "Ball End Milling Analysis and NC Machining of Sculptured Surface Models," *Int. Cont. on Die and Mould*, Vol. 1, No. 1, pp. 218~225.
- (13) Kim, B. H. and Chu, C. N., 1994, "Effect of Cutter Mark on Surface Roughness and Sculop Height in Sculptured Surface Machining," *Computer Aided Design*, Vol. 26, No. 3, pp. 179~188.
- (14) Li, S. X. and Jerard, R. B., 1994, "5-Axis Machining of Sculptured Surfaces with a Flat-End and Ball-End Cutter," *Computer Aided Design*, Vol. 26, No. 3, pp. 165~178.
- (15) Ruegg, A., 1992, "A Generalized Kinematics Model for Three to Five-Axis Milling Machines and Their Implementation in a CNC," *Annals of the CIRP*, Vol. 41., No. 1, pp. 547~550.
- (16) Vickers, G. W. and Quan, K. W., 1989, "Ball-

- Mills Versus End-Mills for Curved Surface Machining," *ASME Journal of Engineering for Industry*, Vol. 115, Nov., pp. 424~431.
- (17) Ralph-Ip, W. L. and Loftus, M., 1992, "Cusp Geometry Analysis in Free-Form Surface Machining," *Int. J. of Production Research*, Vol. 30, No. 11, pp. 2697~2711.
- (18) Joe, B., 1991, "Construction of Three-Dimensional Delaunay Triangulations Using Local Transformations," *Computer Aided Geometric Design*, North-Holland, Vol. 8, pp. 123~142.
- (19) Farin, G., 1990, *Curves and Surfaces for Computer Aided Geometric Design*, Second Edition, Academic Press, Boston, pp. 27~66.
- (20) Andrews, H. C., 1982, *Introduction to Mathematical Techniques in Pattern Recognition*, Wiley-Interscience, New York, pp. 150~164.