

다면체의 회전 스융트 볼륨 계산 방법

백낙훈*, 신성용**

Computing Rotational Swept Volumes of Polyhedral Objects

Nakhoon Baek* and Sung-Yong Shin**

ABSTRACT

Plane sweep plays an important role in computational geometry. This paper shows that an extension of topological plane sweep to three-dimensional space can calculate the volume swept by rotating a solid polyhedral object about a fixed axis. Analyzing the characteristics of rotational swept volumes, we present an incremental algorithm based on the three-dimensional topological sweep technique. Our solution shows the time bound of $O(n^2 \cdot 2^{\alpha(n)} + T_c)$, where n is the number of vertices in the original object and T_c is time for handling face cycles. Here, $\alpha(n)$ is the inverse of Ackermann's function.

Key words : Swept volume, Rotation, Arrangement, Topological sweep

1. 서 론

평면 스융(plane sweep)은 계산기하학(computational geometry)에서 널리 사용되는 방법들 중의 하나이다. 이 방법은 그 이름이 의미하는 대로, 하나의 수직선을 전체 평면의 왼쪽에서 오른쪽으로 이동시키면서 평면 상의 각종 기하학적 형태들을 처리한다. 수행 속도의 향상을 위해, 수직선 대신 위상 직선(topological line)을 사용하는 경우도 있다^[1,2].

2차원 평면에서는 평면 스융에 기초하여 다양한 알고리즘들이 개발되었지만, 3차원 공간에서는 평면 스융에 대응되는 개념인 공간 스융(space sweep)을 이용한 알고리즘들은 아직까지는 그 수가 많지 않다^[3].

본 논문에서는 위상 공간 스융(topological space sweep)을 이용하여, 다면체(polyhedral object)를 고정축을 중심으로 일정한 각도 θ 만큼 회전시켰을 때에 생성되는 회전 스융트 볼륨(rotational swept volume)을 계산할 수 있음을 보인다. 본 논문의 기본 전략은 위상 평면 스융(topological plane sweep)^[4]과 유사하나, 2차원 평면이 아니라 3차원 공간에서 동작한다

는 차이점을 가진다. 평면 스융의 경우는 2차원 평면 상에서 이벤트 포인트(event point)들의 정렬 리스트(sorted list)를 사용하는 반면에 본 논문은 주어진 물체의 경계면(boundary face)들의 위상 정렬 리스트(topologically sorted list)를 사용한다.

3차원 물체가 주어진 시간 간격 동안 공간 상을 이동하면서 지나간 모든 점들의 집합을 스융트 볼륨(swept volume)이라 한다. 스융트 볼륨은 NC 시뮬레이션이나 로봇틱스(robotics) 등의, 다양한 분야에서 응용되고 있다. 예를 들어, NC 가공의 결과는 절삭공구가 지나가면서 생성하는 스융트 볼륨을 계산한 후, 피절삭체와의 집합 연산을 통하여 구할 수 있다^[5,6] 기계 설계나 컴퓨터 애니메이션에서는 물체들 간의 충돌 여부를 검사하기 위해 각 물체의 스융트 볼륨을 이용한다^[7].

주어진 물체의 스융트 볼륨은 물체의 형태(shape)와 이동 경로(path)를 이용하여 계산되어진다. 통상 주어진 물체는 자기 교차(self intersection)가 없는 유효 솔리드(valid solid)라고 가정한다. 이제까지 대부분의 연구 결과들^[5-11]은 임의의 경로를 이동하는 임의의 형태를 가지는 물체들을 대상으로 하였고, 그 결과로 매우 복잡한 연산과 상당한 수행 시간을 요구한다. 일부 연구 결과들^[12,13]은 이산(離散) 형태의

*정회원, 경북대학교 전자전기공학부
**한국과학기술원 전산학과

스윙트 볼륨을 효과적으로 계산할 수 있음을 보였지만, 이들 결과는 NC 시뮬레이션의 화면 출력과 같은 제한된 응용에서만 사용 가능하다.

Martin and Stephenson^[6]과 Weld and Leu^[8]는 원래 물체의 경계면들을 이용하여 스윙트 볼륨을 계산할 수 있음을 보였다. 이들의 결과는 스윙트 볼륨의 특성을 깔끔하게 정리하였지만, 실제 계산에 있어서는 일반적인 집합 연산을 사용하기 때문에 상당한 계산 시간이 필요로 한다.

스윙트 볼륨이 실제로 사용되는 경우에는 대부분의 물체 이동 경로를 평행 이동(translation)이나, 고정축에 대한 회전 이동(rotation)으로 표현(또는 근사)할 수 있다^[14,15]. 본 논문의 저자들은 이미 평행 이동에 의한 스윙트 볼륨을 효과적으로 계산할 수 있음을 보인 바 있다^[16,17]. 평행 이동은 반경이 무한대인 회전 이동과 동일하므로, 이전 연구 결과들은 본 논문이 제시하는 회전 스윙트 볼륨 계산 방법의, 제한된 상황에서의 응용으로 볼 수 있다.

2장에서는 우선 고정축을 중심으로 다면체를 회전시켰을 때의 회전 스윙트 볼륨을 분석한다. 그 결과로 다면체의 최초 위치와 최종 위치에서의 경계면들로부터 회전 스윙트 볼륨의 경계면이 될 수 있는 후보들을 계산할 수 있다. 3장에서는 이 후보 경계면들로부터 최종 스윙트 볼륨을 찾아내는, 위상 공간 정렬에 기초한 알고리즘을 제시한다. 마지막으로 4장에서 결론 및 향후 과제를 제시한다.

2. 스윙트 볼륨의 특성 분석

2.1 유향원에 의한 처리

본 논문에서는 다면체 O 와 그 스윙트 볼륨 SV 를 B-rep(boundary representation)으로 표현한다. 즉, 각 다면체는 경계면들의 집합으로 표현된다^[18]. $B(O)$ 와 $B(SV)$ 를 각각 O 와 SV 의 B-rep 표현이라고 하면, 스윙트 볼륨은 $B(O)$ 로부터 $B(SV)$ 를 계산함으로써 구해진다.

설명 편의를 위해, O 는 Fig. 1에서와 같이, $+z$ 축을 기준으로 반시계 방향으로 회전한다고 가정한다. 임의의 축을 기준으로 하는 회전 이동은 좌표축 변환을 통하여 $+z$ 축 기준의 회전 이동으로 변환시킬 수 있다^[19]. O 는 시간이 흐름에 따라 주어진 경로 상을 이동하고, 시간 t 에서의 O 의 위치는 $O(t)$ 로 표현한다. 시간 t 는 정규화(normalization)시켜 $t \in [0,1]$ 로 가정한다. O 의 시간과 무관한(time-invariant) 특성들에 대해서는 특별히 $O(t)$ 대신 O 로 표현한다.

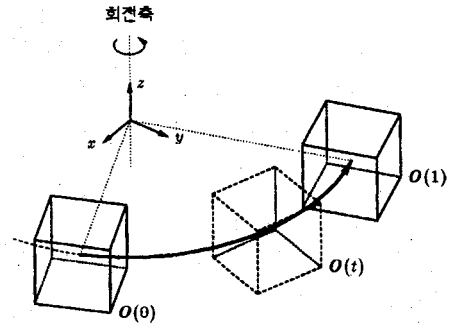


Fig. 1. Rotating an object O about the $+z$ axis.

임의의 점을 $+z$ 축을 기준으로 2π 라디안 이상 회전시켜 얻어지는 원을 유향원(directed circle, 有向圓)이라 하고, 유향원 γ 는 양의 실수 r_1, z_1 을 써서

$$\gamma = \{(x, y, z) \mid x = r_1 \cos \phi, y = r_1 \sin \phi, z = z_1\}$$

로 정의한다. γ 상의 한 점 p 에서 다른 한 점 q 까지의 반시계 방향 경로를 유향 원호(directed arc, 有向圓弧) $[p, q]$ 라 정의한다.

유향원 γ 는 예외 경우(degenerate case)들을 제외하고는 반드시 다면체 O 의 경계면(boundary face)들과 짝수번 교차한다^[11]. 설명의 편의를 위해, 예외 경우들은 이 장의 뒷부분에서 다시 설명한다. γ 상에서 O 의 외부에 속하는 한 점에서 출발하여 γ 의 경로를 따라가면서 O 의 경계면들과 교차하는 점들을 각각 p_i, q_i 라고 하면, Fig. 2(a)에서와 같이, 각각의 p_i, q_i 쌍들은 O 의 내부에 포함되는 유향 원호 $[p_i, q_i]_\gamma$ 들을 형성한다. O 와 교차하는 모든 유향원들의 집합을 $\Gamma = \{\gamma \mid \gamma \cap B(O) \neq \emptyset\}$ 라고 하면,

$$O = \bigcup_{\gamma \in \Gamma} (\Gamma \cap O) = \bigcup_{\gamma \in \Gamma} \bigcup_i [p_i, q_i]_\gamma \quad (1)$$

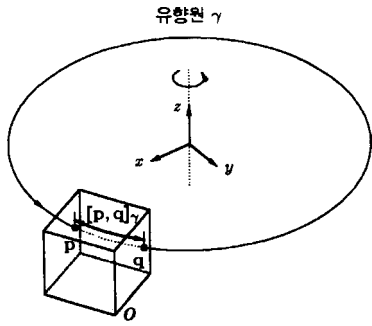
이다. 스윙트 볼륨 SV 는 시간 간격 $T=[0,1]$ 동안의 모든 $O(t)$ 들의 합집합이므로,

$$\begin{aligned} SV &= \bigcup_{t=0}^1 O(t) \\ &= \bigcup_{t=0}^1 \bigcup_{\gamma \in \Gamma} \bigcup_i [p_i(t), q_i(t)]_\gamma \\ &= \bigcup_{t \in \Gamma} \bigcup_i \bigcup_{t=0}^1 [p_i(t), q_i(t)]_\gamma \end{aligned}$$

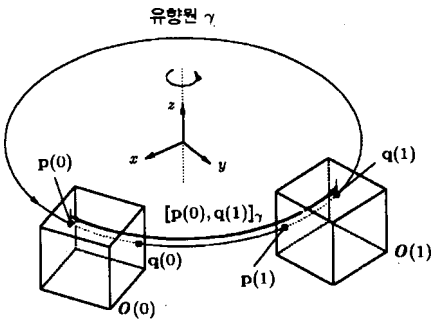
이다. 또, 시간 t 의 경과에 따라 $p_i(t)$ 와 $q_i(t)$ 가 반시계 방향으로 회전하므로, Fig. 2(b)에서와 같이,

$$\bigcup_{t=0}^1 [p_i(t), q_i(t)]_\gamma = [p_i(0), q_i(1)]_\gamma$$

이다. 따라서, $B(SV)$ 는



(a) p_i 's and q_i 's



$$(b) \bigcup_{i=0}^1 [p_i(t), q_i(t)]_\gamma = [p_i(0), q_i(1)]_\gamma$$

Fig. 2. The directed circle γ .

$$B(SV) = B\left(\bigcup_{\gamma \in \Gamma} \bigcup_{i=0}^1 [p_i(t), q_i(t)]_\gamma\right) \\ = B\left(\bigcup_{\gamma \in \Gamma} \bigcup_i [p_i(0), q_i(1)]_\gamma\right)$$

이다.

2.2 면의 분류

식 (1)에서, 유향 원호 $[p_i, q_i]$ 의 양 끝점 p_i, q_i 는 Fig. 3에서와 같이 각각 O 의 경계면(boundary face) f, g 위에 위치한다. 따라서,

$$p_i = \gamma \cap f, \quad q_i = \gamma \cap g \tag{3}$$

이고, 식 (2)를 경계면들을 기준으로 재해석할 수 있

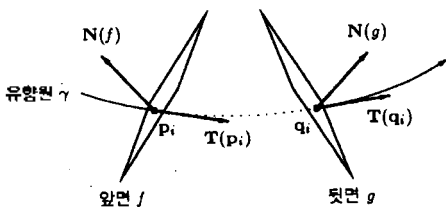


Fig. 3. Tangent vectors and normal vectors.

다. 유향원 γ 의 p_i, q_i 에서의 접선 벡터(tangent vector)들을 각각 $T(p_i), T(q_i)$ 라 하고, 경계면 f, g 의 법선 벡터(normal vector)들을 각각 $N(f), N(g)$ 라 하자. $N(f), N(g)$ 는 각각 O 의 외부 쪽을 가리키므로,

$$T(p_i) \cdot N(f) < 0, \quad T(q_i) \cdot N(g) > 0$$

이다.

유향원 γ 는 하나의 경계면과 2번 교차할 수도 있으므로, 식 (3)의 f 와 g 는 같은 경계면이 될 수도 있다. 이러한 예외적 경우를 피하기 위해, 주어진 경계면들을 단조 경계면(monotone boundary face)들로 분할한다. 단조 경계면으로의 분할 과정은 별도의 참고 문헌^[20]에 자세히 설명되어 있다. 단조 경계면들은 하나의 유향원 γ 와 교차하는 경우에는 반드시 1번 교차하는 성질을 가진다.

단조 경계면 f 에서는 f 의 내부에 속하는 모든 점 p 에 대하여 내적 $T(p) \cdot N(f)$ 가 항상 양이거나 항상 음이 됨을 쉽게 보일 수 있다. 따라서, O 의 단조 경계면들은 두 개의, 서로 소인 집합

$$F^f = \{f^f \mid T(p) \cdot N(f^f) \leq 0 \forall p \in f^f, f^f \in B(O)\}$$

와

$$F^b = \{f^b \mid T(q) \cdot N(f^b) \geq 0 \forall q \in f^b, f^b \in B(O)\}$$

로 분류할 수 있다. 면 $f^f \in F^f$ 와 $f^b \in F^b$ 는 각각 앞면(front face), 뒷면(back face)이라 정의한다. 각 경계면들은 +z축을 기준으로 회전하므로, 위의 분류는 시간 t 와 무관하다. 따라서, 식 (2)에서 $[p_i(0), q_i(1)]_\gamma$ 의 끝점들은

$$p_i(0) = \gamma \cap f^f(0), \quad f^f(0) \in F^f(0)$$

와

$$q_i(1) = \gamma \cap f^b(1), \quad f^b(1) \in F^b(1)$$

로 표현된다.

이제 예외적인 경우들을 살펴보겠다. 앞면 f^f 와 뒷면 f^b 가 공유하는 변(edge)을 윤곽변(silhouette edge)이라 정의한다. 유향원 γ 가 Fig. 4에서와 같이, 윤곽 변 $e^s \subseteq f^f \cap f^b$ 상의 점 p 와 교차할 때, p 는 예외적인

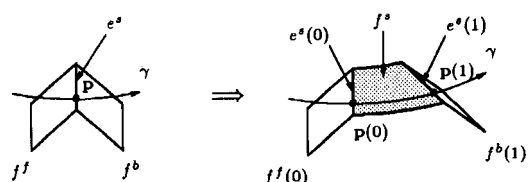


Fig. 4. A silhouette face.

유향 원호 $[p, q]$ 로 볼 수 있다. 이 때, 윤곽면 e^s 는 다음과 같이 정의되는 윤곽면 f^s 를 생성한다.

$$\begin{aligned} f^s &= \bigcup_{t=0}^1 e^s(t) = \bigcup_{p \in e^s} \bigcup_{t=0}^1 [p(t), p(t)]_y \\ &= \bigcup_{p \in e^s} [p(0), p(1)]_y \\ &= [e^s(0), e^s(1)] \end{aligned}$$

여기서, $[e^s(0), e^s(1)]$ 은 $e^s(0)$ 와 $e^s(1)$ 로 둘러싸인 모든 유향 원호들의 집합을 의미한다. 모든 윤곽면들의 집합을

$$E^s = \{e^s \mid e^s \subseteq f^s \cap f^b, f^s \in F^s, f^b \in F^b\}$$

라 하면, 모든 윤곽면들의 집합은

$$F^s = \{f^s \mid f^s = [e^s(0), e^s(1)], e^s \in E^s\}$$

로 정의할 수 있다.

정리하면, $F^s(0), F^b(1), F^s$ 가 $B(SV)$ 의 후보(candidate)가 될 수 있으므로,

$$B(SV) \subseteq (F^s(0) \cup F^b(1) \cup F^s)$$

이다. $F^s(0) \cup F^b(1) \cap F^s$ 에 속하는 하나의 면 f 는 SV 의 내부에 일부(또는 전체)가 포함될 수 있다. 이러한 내부에 속하는 부분들은 $B(SV)$ 의 후보들에서 제외되어야 한다.

2.3 면 단위 처리

$F^s(0) \cup F^b(1) \cap F^s$ 로부터 $B(SV)$ 를 계산하기 위해 은면 제거 방법^[19]을 사용할 수 있다. 모든 후보면들을 3차원 공간에 배치한 후, 각각의 후보면들이 SV 의 외부에서 보이는지를 검사한다. 이 때, 면 f 가 항상 다른 면들에 의해 가려지면, f 는 스윙트 볼륨의 내부에 위치하고, $B(SV)$ 에는 당연히 속하지 않게 된다. 본 논문에서는 이러한 관점에서 출발하여, 기존의 우선 순위(priority)에 의한 은면 제거 방법들^[21,22]에서와 같이, 면들을 위상적으로 정렬한 후 스윙트 볼륨을 계산한다.

유향원 γ 상의 점 $p=(x, y, z)$ 에서, $\phi(p)$ 는 γ 의 반지름 r_1 에 대하여

$$x = r_1 \cos \phi, y = r_1 \sin \phi$$

를 만족시키는 각 ϕ 로 정의한다. $t=0$ 일 때, 각 ϕ 를 $[0, 2\pi]$ 로 정규화 시키면, $B(O(0))$ 에 속하는 모든 점 $p(0)$ 는 $\phi(p(0)) \in [0, 2\pi]$ 를 만족한다. 또, O 의 회전 각도를 θ 라 하면, $B(O(1))$ 에 속하는 모든 점 $p(1)$ 에서는 $\phi(p(1)) \in [\theta, \theta+2\pi]$ 이다. 유향원 γ 상의 두 점들 간

의 우선 순위 관계는 각 ϕ 를 이용하여 다음과 같이 정의한다.

정의 1. (점들간의 우선 순위) 유향원 γ 상의 두 점 p, q 가 $\phi(p) \leq \phi(q)$ 를 만족시키면, 점 p 는 점 q 에 우선한다고 하고, $p \leq_\gamma q$ 로 표시한다.

두 경계면 f_i, f_j 를 모두 교차하는 모든 유향원들의 집합을 $\Gamma_{q_i, \beta} = \{\gamma \mid \gamma \cap f_i \neq \emptyset, \gamma \cap f_j \neq \emptyset\}$ 로 표시하면, 두 경계면들 간의 우선 순위는 다음과 같이 정의한다.

정의 2. (면들간의 우선 순위) $\Gamma_{q_i, \beta} \neq \emptyset$ 이고, $\Gamma_{q_i, \beta}$ 에 속하는 모든 유향원 γ 에 대하여 $(\gamma \cap f_i) \leq_\gamma (\gamma \cap f_j)$ 이면, f_i 는 f_j 에 우선한다고 하고, $f_i \leq f_j$ 로 표시한다.

실제로는 면들간의 우선 순위를 정하기 위한 전처리 단계가 필요하다. ϕ 값이 0인 모든 점들은 xz -평면 상의 $x \geq 0$ 인 부분으로 이루어진 하나의 반평면(half-plane)을 형성한다. 이 반평면을 특별히 기준 반평면(reference half-plane) H_{ref} 라 하자. 경계면 f 가 H_{ref} 와 교차한다면, Fig. 5에서와 같이, f 는 H_{ref} 를 기준으로 두 개의 영역 f_+ 와 f_- 로 나뉘어진다. f_+ 에 속하는 점 p 에서는 $\phi(p)$ 가 0에 가까운 값이고, f_- 에 속하는 점 q 에서는 $\phi(q)$ 가 2π 에 가까운 값이다. 또다른 면 g 로 보아서는 $\gamma \in \Gamma_{(f_+, g)}$ 인 경우에 $(\gamma \cap f) \leq_\gamma (\gamma \cap g)$ 이고, $\gamma \in \Gamma_{(f_-, g)}$ 인 경우에는 $(\gamma \cap g) \leq_\gamma (\gamma \cap f)$ 이므로, 일관성을 잃게 된다. 이 문제를 해결하기 위해, 경계면이 H_{ref} 와 교차하는 경우에는 H_{ref} 를 기준으로 2개의 면으로 분리한다.

O 가 유효 솔리드(valid solid)이므로, $B(O)$ 에 속하는 두 경계면 f_i, f_j 는 둘 사이의 공유면을 제외하고는 서로 교차할 수 없다. 따라서, 하나의 유향원이 f_i, f_j

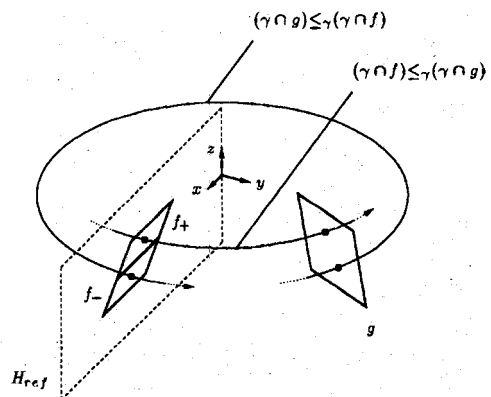


Fig. 5. Reference half-plane H_{ref} .

모두와 교차한다면 $f_i \leq f_j$ 이거나, $f_i \leq f_j$ 의 두 우선 순위 관계 중 오직 하나만을 만족시킨다. 두 경계면 f_b, f_f 간의 우선 순위 관계는 $\Gamma_{(f_b, f_f)}$ 에 속하는 유향된 γ 와의 교차점 $(\gamma \cap f_i), (\gamma \cap f_j)$ 간의 우선 순위를 검사하여 결정한다. 일부 (f_b, f_f) 쌍에서는 $\Gamma_{(f_b, f_f)} = \emptyset$ 이므로, 경계면들 간의 우선 순위 관계는 부분 순서(partial order)이다. 이 부분 순서들은 경계면들과 그들의 우선 순위 관계를 각각 노드(node)와 유향 에지(directed edge)로 하는 그래프 G 로 표현할 수 있다. 그래프 G 를 위상 정렬(topological sort)하여 전체 경계면들에 대한 리스트

$$L = (f_1, f_2, f_3, \dots, f_n)$$

을 계산할 수 있고, f_i 에 우선하는 모든 면들은 리스트 L 상에서 f_i 보다 앞쪽에 위치한다.

이러한 모든 면들의 위상 정렬 역시 우선 순위를 이용한 은면 제거 방법들 중의 일부에서 사용하고 있다¹⁰⁾. 그래프 G 는 이들 방법들에서와 마찬가지로, 경우에 따라 사이클(cycle)을 가질 수 있고, 사이클이 존재하는 그래프 G 는 위상 정렬의 직접적인 적용이 불가능하다. 사이클이 존재하는 그래프 G 에서는 $w \rightarrow \dots \rightarrow u$ 로의 경로(path)가 존재하는 유향 에지

$u \rightarrow w$ 가 하나 이상 존재한다. 이 때, 노드 u, w 에 대응되는 경계면들을 각각 f_u, f_w 라고 하면, f_u 를 $f_{u1} \leq f_w$ 와 $f_w \leq \dots \leq f_{u2}$ 를 만족시키는 2개의 면 f_{u1}, f_{u2} 로 분리시키면, 유향 에지 $u \rightarrow w$ 가 제거되고, 결과적으로 해당 사이클을 제거할 수 있다.

강한 우선 순위 관계는 리스트 L 을 이용하여 다음과 같이 정의한다.

정의 3. (강한 우선 순위) $f_i \leq f_j$ 를 만족시키는 두 경계면 f_i, f_j 에 대하여, $f_i \leq f_k \leq f_j$ 를 만족시키는 다른 경계면 f_k 가 존재하지 않으면, f_i 는 f_j 에 대하여 강한 우선 순위를 가진다고 하고, $f_i \leq f_j$ 로 표시한다.

뒷면 f^b 와 앞면 f^f 가 $f^b \leq f^f$ 를 만족시키면, Fig. 6(a)에서와 같이 $q_b \in (\gamma \cap f^b), p_{i+1} \in (\gamma \cap f^f)$ 이고, 서로 인접한 두 개의 유향 원호 $[p_i, q_b]_p, [p_{i+1}, q_{i+1}]_p$ 를 쉽게 찾을 수 있다. 이 때,

$$\begin{aligned} q_b(0) &= \gamma \cap f^b(0), \\ q_b(1) &= \gamma \cap f^b(1), \\ p_{i+1}(0) &= \gamma \cap f^f(0) \end{aligned}$$

이므로, $f^b(1)$ 와 $f^f(0)$ 상에서 $B(SV)$ 에 속하는 부분들을

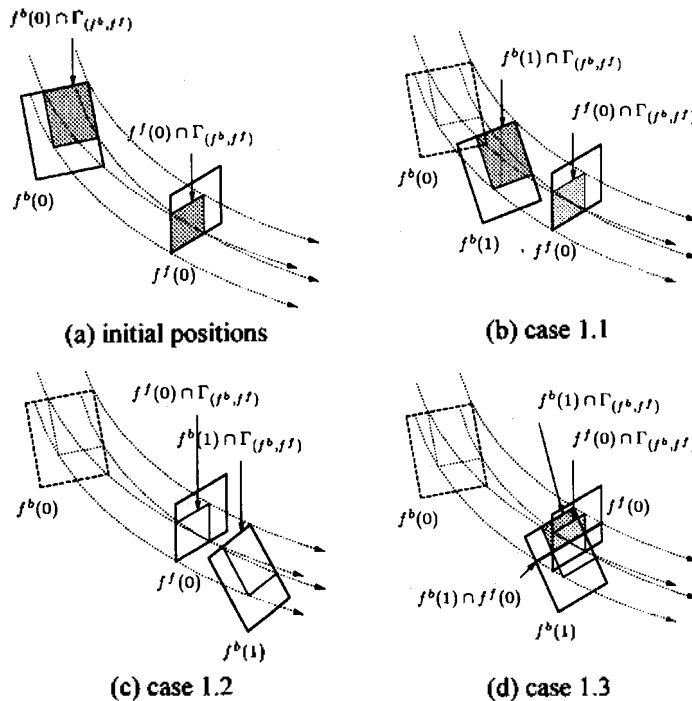


Fig. 6. Front and back faces.

아래와 같이 3개의 경우로 나누어 찾아낼 수 있다.

경우 1.1 ($f^b(0) \leq f^b(1) \leq f^f(0)$): Γ_{f^b, f^f} 에 속하는 유향원 γ 는 $f^b(1)$ 와 교차하면서 SV의 외부로 나가고, $f^f(0)$ 를 교차하면서 다시 SV의 내부로 들어온다. 따라서,

$$f^b(1) \cap \Gamma_{(f^b, f^f)} \in B(SV),$$

$$f^f(0) \cap \Gamma_{(f^b, f^f)} \in B(SV)$$

이다(Fig. 6(b)).

경우 1.2 ($f^b(0) \leq f^f(0) \leq f^b(1)$): Γ_{f^b, f^f} 에 속하는 유향원 γ 는 항상 SV의 내부에 포함된다. 따라서,

$$f^b(1) \cap \Gamma_{(f^b, f^f)} \notin B(SV),$$

$$f^f(0) \cap \Gamma_{(f^b, f^f)} \notin B(SV)$$

이다(Fig. 6(c)).

경우 1.3 ($f^b(1) \cap f^f(0) \neq \emptyset$): $f^b(1)$ 과 $f^f(0)$ 가 각각 다각형이므로, 이들의 교집합은 직선 l 에 속하는 선분이다. l 을 기준으로 각각은 다시

$$f^b(0) \leq f^b(1) \leq f^f(0),$$

$$f^f(0) \leq f^f(1) \leq f^b(1)$$

의 두 경우로 나뉘고, 각각은 경우 1.1과 경우 1.2를 이용하여 처리한다(Fig. 6(d)).

또, 윤곽면 $f^s = [e^s(0), e^s(1)]$ 상에서는 윤곽면 e^s 를 분석하여 $B(SV)$ 에 속하는 부분을 계산한다. 주어진 e^s 에 대하여, $f^b \leq e^s \leq f^f$ 를 만족시키는 뒷면 f^b 와 앞면 f^f 를 각각 찾을 수 있다. $f^b(0) \leq e^s(0) \leq f^b(1)$ 이면, $[e^s(0), f^b(1)]$ 는 SV의 내부에 속하므로, $[e^s(0), f^b(1)] \notin B(SV)$ 이다. 마찬가지로, $f^f(0) \leq e^s(1) \leq f^f(1)$ 이면, $[f^f(0), e^s(1)] \notin B(SV)$ 이다. 따라서, $B(SV)$ 에 속하는 $[e^s(0), e^s(1)]$ 상의 부분은 Fig. 7에서와 같이,

$$[e^s(0), e^s(1)] - [e^s(0), f^b(1)] - [f^f(0), e^s(1)] \quad (4)$$

이다.

3. 알고리즘

본 논문이 제시하는, 회전 스윙트 볼륨을 계산하는 알고리즘은 평면 상에 사영(project)된 경계면들의 어레인지먼트(arrangement)^[1,2]에 기초하므로, 경계면들의 회전 사영(rotational projection)을 미리 정의한다. 기하학적 형태 E 와 교차하는 모든 유향원들의 집합을 $\Gamma_E = \{\gamma \mid \gamma \cap E \neq \emptyset\}$ 라 하면, E 의 H_{ref} 위로의

$$[e^s(0), e^s(1)] - [e^s(0), f^b(1)] - [f^f(0), e^s(1)]$$

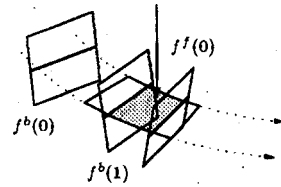


Fig. 7. Processing a silhouette face.

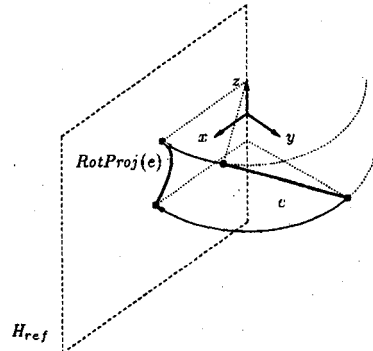


Fig. 8. RotProj (e).

회전 사영 결과는

$$RotProj(E) = \bigcup_{\gamma \in \Gamma_e} (\gamma \cap H_{ref})$$

이다. 면 f 의 경우, $RotProj(f)$ 는 H_{ref} 상의 2차원 폐영역이 된다. f 는 $+z$ 축을 중심으로 회전하므로, $RotProj(f)$ 는 시간 t 와는 무관하다.

f 의 경계를 정의하는 에지들 중의 하나를 e 라 하면, e 는 (x_1, y_1, z_1) 에서 (x_2, y_2, z_2) 까지의 선분이다. 이때, e 는 파라미터 $u \in [0, 1]$ 을 이용하여

$$e(u) = (x_1 + u(x_2 - x_1), y_1 + u(y_2 - y_1), z_1 + u(z_2 - z_1))$$

로 표현할 수 있다. Fig. 8에서와 같이, e 의 회전 사영 결과는

$$RotProj(e(u)) = (\sqrt{(x_1 + u(x_2 - x_1))^2 + (y_1 + u(y_2 - y_1))^2}, 0, z_1 + u(z_2 - z_1))$$

이다. $e(u)$ 의 회전 사영 결과는 Table 1에서와 같이, H_{ref} 상의 직선 또는 2차 쌍곡선의 일부분이다. 따라서, $RotProj(f)$ 는 선분 또는 2차 쌍곡선의 일부분들로 둘러 싸인다.

$RotProj(f)$ 들의 어레인지먼트를 점진적으로 구성해 나가면서 서로 인접한 경계면들을 찾아낼 수 있다.

Table 1. Geometric Shape of $RotProj(e)$

e 의 기하학적 성질	$RotProj(e)$ 을 포함하는 도형
$z_1 = z_2$	직선 $z = z_1$
$x_1 = x_2$ and $y_1 = y_2$	직선 $x = \sqrt{x_1^2 + y_1^2}$
$x_1 z_2 = x_2 z_1$ and $y_1 z_2 = y_2 z_1$	꺼은 선 $x = \pm \frac{b}{a} z$
$x_1 y_2 = x_2 y_1$	꺼은 선 $x = \pm \frac{b}{a} z - p $
나머지 경우들	2차 쌍곡선 $a^2 x^2 = b^2 (z - p)^2 + q^2$

L 에서 최초 $(j-1)$ 개의 경계면 $f_i, 1 \leq i \leq j-1$ 들이 H_{ref} 위에 회전 사영 되어 있다고 가정하자. $RotProj(f_i), 1 \leq i \leq j-1$ 의 각 변들은 H_{ref} 상에서 직선 또는 2차 쌍곡선들의 일부분들로 이루어진 어레인지먼트 A_{j-1} 을 형성한다.^[23] 이 어레인지먼트 내의 영역 $a_k \in A_{j-1}$ 의 유효면은 다음과 같이 정의한다.

정의 4. (유효면) 면 $f_p, 1 \leq p \leq j-1$ 에 대하여, 영역 $a_k \in A_{j-1}$ 이 $a_k \subseteq RotProj(f_p)$ 를 만족시키고, $f_p \subseteq f_b, a_k \subseteq RotProj(f_j)$ 를 만족시키는 면 $f_b, 1 \leq i \leq j-1$ 이 존재하지 않을 때, f_p 를 영역 a_k 의 유효면이라 한다.

즉, 유효면은 L 상에서 최초 $(j-1)$ 개의 경계면들 중 $a_k \in A_{j-1}$ 를 마지막으로 수정한 면이다. 따라서, $RotProj(\gamma) \in a_k$ 인 모든 유향원 γ 에 대하여 $\gamma \cap f_p \neq \emptyset$ 이다.

다음 단계에서는 L 상의 j 번째 면 f_j 로부터 $RotProj(f_j)$ 를 계산하여 A_{j-1} 로부터 새로운 어레인지먼트 A_j 를 계산한다. $RotProj(f_j)$ 의 내부에 영역 $a_k \in A_{j-1}$ 의 일부(또는 전체)가 포함된다고 가정하자. 이 때, a_k 는 $a_k^r = a_k \cap RotProj(f_j)$ 와 $a_k^{qu} = a_k - a_k^r$ 의 두 부분으로 나누어진다. a_k 의 유효면을 f_p 라 하고, $\Gamma_{a_k^r} = \{\gamma \mid RotProj(\gamma) \in a_k^r\}$ 이면, $\gamma \in \Gamma_{a_k^r}$ 인 모든 유향원 γ 는 f_p 와 교차한다. 또, $a_k^r \subseteq RotProj(f_j)$ 이므로, $\Gamma_{a_k^r}$ 에 속하는 모든 γ 에 대하여 $\gamma \cap f_p \neq \emptyset$ 이다. 따라서, $\Gamma_{a_k^r}$ 에 속하는 모든 γ 는 f_p, f_j 모두와 교차하고,

$$\Gamma_{a_k^r} \subseteq \Gamma_{(f_p, f_j)} \tag{5}$$

이다. f_p 가 a_k 의 유효면이므로, $f_p \subseteq f_i \subseteq f_j$ 를 만족시키는 f_i 는 존재하지 않는다. 따라서,

$$f_p \leq f_j \tag{6}$$

이다.

이 때, j 번째 면 f_j 는 앞면일 수도 있고 뒷면일 수도 있으므로, 아래의 2가지 경우가 가능하다.

경우 2.1 (j 번째 면이 앞면 f_j 인 경우): O 가 유효 솔리드이므로, a_k 의 유효면은 뒷면 f_b 이다. 식 (5), (6)에 의해, $\Gamma_{a_k^r} \subseteq \Gamma_{(f_p, f_j)}, f_b \leq f_j$ 이다.

따라서, 2장에서와 같은 결과에 따라, 각각의 영역 $a_k^r \subseteq RotProj(f_j)$ 에서 $f_b(1), f_j(0)$ 상에서의 SV 의 경계면들을 추출할 수 있다.

경우 2.2 (j 번째 면이 뒷면 f_j 인 경우): 이후의 어레인지먼트 수정 중에 f_j 에 대응되는 앞면 $f_i, i > j$ 가 찾아질 것이다. 그러한 f_i 를 찾을 때까지 처리를 미룬다.

어느 경우에서든 다음 단계들에서의 처리를 위하여 $a_k^r \subseteq RotProj(f_j)$ 의 유효면을 갱신한다. 이 때, a_k^r 들을 유효면으로 하는 하나의 영역 a_r 로 합친다. 따라서, A_j 에서는 유효면이 f_j 인 영역은 a_r 뿐이다. 남은 부분들인 a_k^{qu} 들은 원래의 유효면을 그대로 유지한다. 이러한 영역들 간의 합병은 어레인지먼트 A_j 의 관리를 단순화시킨다.

L 의 앞부분에 위치하는 일부 앞면들은 대응되는 뒷면이 없을 수도 있다. 영역 a_k^r 에서 앞면 f' 에 대응되는 뒷면이 없다고 가정하자. 유향원들이 $B(O)$ 와 짝수번 만나므로, L 의 뒷부분에 위치하는 뒷면(또는 뒷면들) f'' 는 대응되는 앞면이 없는 대신, $\Gamma_{a_k^r} = \{\gamma \mid RotProj(\gamma) \in a_k^r\}$ 에 속하는 유향원들과 만난다. 경우 1.1, 1.2, 1.3에서 보인 바와 같이, SV 의 경계면들은 유향원들을 기준으로 판별한다. $\Gamma_{a_k^r}$ 에 속하는 유향원을 f'' 상의 한 점에서 출발하여 추적하면, f'' 는 $(f' \cap \Gamma_{a_k^r})$ 와 대응된다. 따라서, 이러한 들을 찾으면, L 의 마지막에 첨가시켜 해당 f'' 들과 대응되도록 한다.

경우 2.1, 2.2에 따라, 모든 경계면들을 L 상의 순서대로 처리하여 점진적으로 어레인지먼트를 수정함으로써 전체 $B(SV)$ 를 계산할 수 있다. 회전 스윕트 볼륨을 계산하는 *RotationalSweptVolume* 알고리즘은 Fig. 9와 같다. 이 알고리즘의 시간 복잡도를 계산하기 위해, O 의 꼭지점 수를 n 이라 하자. 편의상 O 의 모든 면들이 삼각형이라고 가정하면, O 의 변 수와 면 수는 모두 $O(n)$ 이다.

단계 1은 경계면들을 위상 정렬하기에 앞선 전처리 과정이다. 이 단계에서의 모든 연산은 $O(n)$ 시간

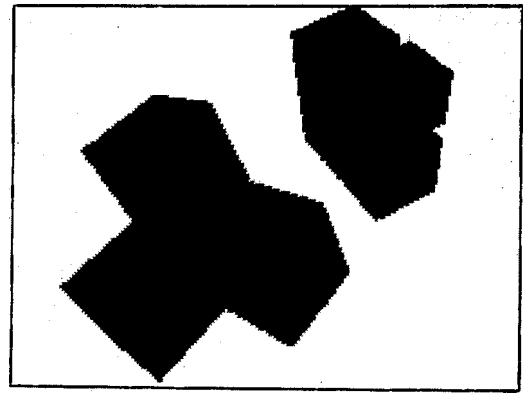
```

Algorithm RotationalSweptVolume
  input: 물체  $O$ , 회전 이동에 대한 정보  $M$ 
  output: 스윕트 볼륨  $SV$ 
Object function RotationalSweptVolume(Object  $O$ ,
  Motion  $M$ ) begin
  { 단계 1: 전처리 과정 }
   $O$ 의 경계면들을  $H_{xy}$ 를 기준으로 분할;
   $O$ 의 경계면들을 단조 경계면들로 분할;
   $O$ 의 경계면들을  $F^f$ 와  $F^b$ 로 분류;
   $O$ 의 변들에서 윤곽면들을  $E^f$ 로 분류;
  { 단계 2: 위상 정렬 }
   $L \leftarrow (F^f \cup F^b)$ 의 면들을 위상 정렬한 리스트;
  { 단계 3:  $B(SV)$  계산 }
  Arrangement  $A \leftarrow \emptyset$ ;
  for each Face  $f \in L$  do
     $A \leftarrow A \cup RotProj(f)$ ;
     $R \leftarrow RotProj(f)$ 의 내부에 속하는 영역  $a_i^n$ 들;
    if  $f \in F^f$  then
      for each Region  $a \in R$  do
         $g \leftarrow a$ 의 유효면;
        if  $g = \emptyset$  then
           $(f \cap \Gamma_a)$ 를  $L$ 의 끝부분에 첨가;
        else
           $g$ 와  $f$ 상의  $B(SV)$ 에 속하는
          부분들을 경우 1.1, 1.2, 1.3에
          따라 계산;
          식 (4)에 따라, 윤곽면 상의  $B(SV)$ 에
          속하는 부분 계산;
        endif
      end { for }
    endif
     $R$ 에 속하는 모든 영역들을 새로운 영역  $a$ ,로
    합치고, 그 유효면을  $f$ 로 바꿈;
  end { for }
  {  $SV$  반환 }
   $SV \leftarrow B(SV)$ 로 표현되는 솔리드;
  return  $SV$ ;
end { function }
  
```

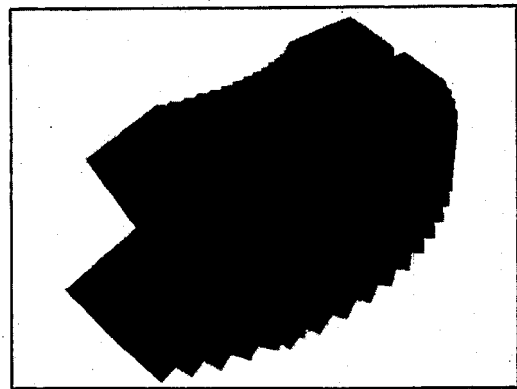
Fig. 9. The algorithm *RotationalSweptVolume*.

에 처리됨을 쉽게 보일 수 있다. 단계 2에서는 O 의 경계면들을 그 우선 순위 관계에 따라 위상 정렬한다. 2장에서 보인 바와 같이, 이 위상 정렬 과정은 우선 순위에 기초한 은면 제거 방법들에서와 유사하다. 따라서, 우선 순위들 사이에 생길 수 있는 싸이클을 다루는 데에 걸리는 시간을 t_c 라 하면, 단계 2의 시간 복잡도는 $O(n^2 + t_c)$ 이다.

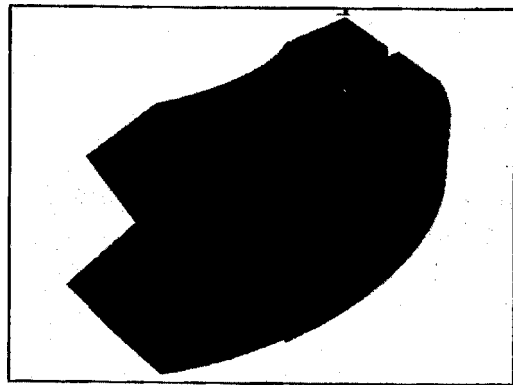
단계 3은 위상 정렬 리스트 L 에 주어진 경계면들의 순서에 따라, 어레인지먼트 A 를 점진적으로 계산하여 $B(SV)$ 를 생성한다. 어레인지먼트 A 내의 2차 곡



(a) $O(0)$ and $O(1)$



(b) sixteen in-between $O(t)$'s



(c) the swept volume

Fig. 10. An example rotational swept volume.

선들의 수가 $O(n)$ 이므로, A 의 형성에는 $O(n^2 \cdot 2^{\alpha(n)})$ 시간이 걸리고, 모두 $O(n^2)$ 개의 영역이 형성될 수 있다.^{123,24} 여기서, $\alpha(n)$ 은 Ackermann 함수의 역함수이다. 어레인지먼트의 점진적인 계산을 제외한 나머지 처리 과정은 어레인지먼트 내의 영역 수에 비례하

로, $O(n^2)$ 시간이 걸린다. 따라서, 단계 3은 모두 $O(n^2 \cdot 2^{\alpha n})$ 시간을 필요로 한다.

종합하면, *RotationalSweptVolume* 알고리즘은 경계면들의 우선 순위 관계에 사이클이 없다면, $O(n^2 \cdot 2^{\alpha n})$ 의 시간 복잡도를 가진다. 사이클이 있는 경우에는 사이클을 제거하고, 사이클의 제거에 따라 새로 생긴 면들을 처리하는 데에 추가 시간이 필요하다. 이러한 추가 시간을 T_c 라고 하면, 전체 알고리즘은 $O(n^2 \cdot 2^{\alpha n} + T_c)$ 시간에 수행된다. 2장에서 이미 언급한 바와 같이, 일부 면면 제거 방법들^[21,22]도 이러한 사이클 처리를 필요로 한다. 이들 방법들에서 이미 보인 바와 같이, 나선 형태로 삼하게 꼬인 물체를 제외하고는 이러한 사이클이 생기는 경우는 매우 드물다.

공간 복잡도의 측면에서는 어레이지먼트 A 와 그래프 G 의 저장에 각각 $O(n^2)$ 의 공간이 필요하다. O 와 SV 의 저장에 필요한 공간들 역시 $O(n^2)$ 로 제한되므로, 전체 공간 복잡도는 $O(n^2)$ 이다. Fig. 10은 회전 스윕트 볼륨 계산의 한 예이다.

4. 결 론

본 논문은 3차원 공간에서의 위상 스윕 방법으로 회전 스윕트 볼륨을 $O(n^2 \cdot 2^{\alpha n} + T_c)$ 시간에 계산할 수 있음을 보였다. 여기서, n 은 원래 물체의 꼭지점 수이고, T_c 는 경계면들의 정렬 시에 생길 수 있는 사이클의 처리에 걸리는 시간이다. 이전까지의 연구 결과들과는 달리, 본 논문에서는 알고리즘 내의 자세한 처리 방법들을 다루었고, 그 시간 복잡도를 계산하였다.

이전의 연구 결과들^[6,17]에서는 평행 이동에 의한 스윕트 볼륨을 $O(n^2 \cdot \alpha(n) + T_c)$ 시간에 계산하는 알고리즘을 제시하였다. 이전의 알고리즘과 본 논문이 새로 제시하는 알고리즘을 결합하면, 좀더 다양한 형태의 이동에 대한 스윕트 볼륨 계산이 가능해 질 것이다. 예를 들어, 일련의 회전 및 평행 이동에 의한 스윕트 볼륨은 두 알고리즘의 개별적 결과들을 집합 연산으로 결합하여 구할 수 있다.

감사의 글

본 논문은 과학 재단의 박사 후 해외 연수 과정 지원 사업, 인공지능연구센터(CAIR)의 ERC 프로그램, 정보통신부의 교수 요원 풀 지원 사업의 지원을 받은 결과이다.

참고문헌

1. Shamos, M.I., *Computational Geometry*, Springer Verlag, New York, 1979.
2. O'Rourke, J., *Computational Geometry in C*, Cambridge University Press, 1993.
3. Hertel, S., Mäntylä, M., Mehlhorn, K. and Nievergelt, J., "Space sweep solves intersection of convex polyhedra", *Acta Informatica*, Vol. 21, No. 5, pp. 501-519, 1985.
4. Edelsbrunner, H. and Guibas, L.J., "Topologically sweeping an arrangement", *J. Comput. Syst. Sci.*, Vol. 38, pp. 165-194, 1989.
5. Wang, W.P. and Wang, K.K., "Geometric modeling for swept volume of moving solids", *IEEE Computer Graphics and Applications*, Vol. 6, No. 12, pp. 8-17, Dec. 1986.
6. Martin, R.R. and Stephenson, P.C., "Sweeping of three-dimensional objects", *Computer-Aided Design*, Vol. 22, No. 4, pp. 223-234, May 1990.
7. Kieffer, J. and Litvin, F.L., "Swept volume determination and interference detection for moving 3-D solids", *ASME Journal of Mechanical Design*, Vol. 113, pp. 456-463, Dec. 1991.
8. Weld, J.D. and Leu, M.C., "Geometric representation of swept volumes with application to polyhedral objects", *The International Journal of Robotics Research*, Vol. 9, No. 5, pp. 105-117, Oct. 1990.
9. Blackmore, D. and Leu, M.C., "Analysis of swept volume via Lie groups and differential equations", *The International Journal of Robotics Research*, Vol. 11, No. 6, pp. 516-537, Dec. 1992.
10. Blackmore, D., Leu, M.C. and Shih, F., "Analysis and modelling of deformed swept volumes", *Computer-Aided Design*, Vol. 26, No. 4, pp. 315-326, Apr. 1994.
11. Sourin, A.I. and Pasko, A.A., "Function representation for sweeping by a moving solid", *IEEE Trans. Visualizat. Comput. Graph.*, Vol. 2, No. 1, pp. 11-18, Mar. 1996.
12. Van Hook, T., "Real-time shaded NC milling display", *Computer Graphics (SIGGRAPH '86 Proceedings)*, Vol. 20, pp. 15-20, Aug. 1986.
13. Hui, K.C., "Solid Sweeping in image space-application in NC simulation", *Visual Computer*, Vol. 10, No. 6, pp. 306-316, 1994.
14. Childs, J.J., *Numerical Control Part Programming*, Industrial Press Inc., New York, N.Y., 1973.
15. Elber, G., "Accessibility in 5-axis milling environment", *Computer-Aided Design*, Vol. 26, No.

11, pp. 796-802, Nov. 1994.

16. 백낙훈, 신성용, "평행 이동에 의한 스윕트 볼륨의 계산 방법", 한국CAD/CAM학회 논문집, Vol. 2, No. 1, pp. 28-34, Mar. 1997.

17. Baek, N., Shin, S.Y. and Chwa, K.Y., "On computing translational swept volumes", *International Journal of Computational Geometry & Applications*, (accepted).

18. Mäntylä, M.J., *An Introduction to Solid Modeling*, Computer Science Press, Rockville, MD, 1988.

19. Foley, J.D., van Dam, A., Feiner, S.K. and Hughes, J.F., *Computer Graphics, Principles and Practice, Second Edition*, Addison-Wesley, Reading, Massachusetts, 1990.

20. Baek, N. and Shin, S.Y., "On circularly-hidden surface removal", *Information Processing Letters*, Vol. 66, No. 3, pp. 119-123, May 1998.

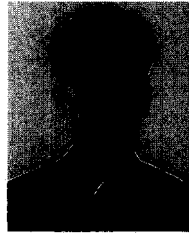
21. McKenna, M., "Worst-case optimal hidden-surface removal", *ACM Transactions on Graphics*, Vol. 6, pp. 19-28, 1987.

22. Schmitt, A., "Time and space bounds for hidden line and hidden surface algorithms", *Proc. Eurographics '81*, pp. 43-56, Amsterdam, Netherlands, 1981.

23. Edelsbrunner, H., Guibas, L., Pach, J., Pollack, R., Seidel, R. and Sharir, M., "Arrangements of curves in the plane: Topology, combinatorics, and algorithms", *Theoret. Comput. Sci.*, Vol. 92, pp. 319-336, 1992.

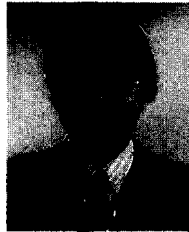
24. Wiernik, A. and Sharir, M., "Planar realizations of

nonlinear Davenport-Schinzel sequences by segments", *Discrete Comput. Geom.*, Vol. 3, pp. 15-47, 1988.



백 낙 훈

1990년 한국과학기술원 과학기술대학
전산학 학사
1992년 한국과학기술원 전산학 석사
1997년 한국과학기술원 전산학 박사
1997년 ~ 1998년 George Washington
University EECS Post-Doc
1998년 ~ 현재 경북대학교 전자전기공학
부 조빙교수
관심분야 : Computer Graphics, Com-
putational Geometry, Geome-
tric Modeling



신 성 용

1970년 한양대학교 산업공학과 학사
1983년 University of Michigan 산업공학
석사
1986년 University of Michigan 산업공학
박사
1987년 ~ 현재 한국과학기술원 전산학과
교수
관심분야 : Computer Graphics, Algori-
thm Design and Analysis,
Computer-Aided Geometric
Design