

토폴로지와 수치적 정확도를 통합한 기하모델링에 관한 연구: 곡면간 교차선

고 광 희*

A Study on Unifying Topology and Numerical Accuracy in Geometric Modeling: Surface to Surface Intersections

Kwanghee Ko*

ABSTRACT

In this paper, we address the problem of robust geometric modeling with emphasis on surface to surface intersections. We consider the topology and the numerical accuracy of an intersection curve to find the best approximation to the exact one. First, we perform the topological configuration of intersection curves, from which we determine the starting and ending points of each monotonic intersection curve segment along with its topological structure. Next, we trace each monotonic intersection curve segment using a validated ODE solver, which provides the error bounds containing the topological structure of the intersection curve and enclosing the exact root without a numerical instance. Then, we choose one approximation curve and adjust it within the bounds by minimizing an objective function measuring the errors from the exact one. Using this process, we can obtain an approximate intersection curve which considers the topology and the numerical accuracy for robust geometric modeling.

Key words : surface to surface intersection, geometric modeling, robustness, validated ODE solver, topology, accuracy

1. 서 론

컴퓨터가 일반화된 지 20여 년이 지난 지금, 산업 현장에서는 보다 복잡하고 높은 성능을 가진 상품의 개발을 위해, 모델링, 시뮬레이션, 해석과 최적화를 위한 각종 소프트웨어를 필수적으로 사용하고 있다. 이러한 다양한 소프트웨어의 체계적이고 종합적인 사용을 통해서 디지털 실험실(digital laboratory)이나 디지털 생산(digital manufacturing)을 구현하는 단계에 있으며 이러한 가상의 환경하에서 상품의 성능 분석을 통한 디자인이나, 또는 시뮬레이션을 통한 생산 단계의 최적 여부를 평가하는 것이 가능하게 되었다. 특히 디지털 실험실은, 시간과 비용이 많이 들어가는 실제 모형 제작을 통한 디자인의 평가 대신, 다양한 설계에 대한 아이디어를 가상의 컴퓨터 모델을 이용해서 쉽

고 빠르게 알아볼 수 있는 바탕을 마련해 주며, 작은 변화에 따른 디자인의 성능 특성 변화의 확인을 통해서 최적 디자인을 보다 빠르게 찾는 것을 가능하게 하였다.

이러한 디지털 실험실 환경을 통한 개발은 정확한 기하 모델로부터 시작하며 그 중심에는 그러한 모델을 생성 하는 CAD기술이 있다. CAD에 관한 연구가 시작 된지 30여 년이 지난 지금, 여러 상용 CAD 프로그램들이 개발되었고 그 프로그램들은 학교 및 산업 현장에서 없어서는 안될 도구로 사용되고 있다. 그렇지만 이러한 비약적인 발전에도 불구하고, 현재 사용되는 CAD 시스템이 실제 제품의 설계와 개발에서 사용될 때 기대된 수준만큼 도달하지 못한 것도 사실이다. 이러한 어려움의 근간에는 그 동안 간과되었던 여러 가지 수학적인 문제들이 포함되어 있으며, 기하 모델의 생성에 관련된 계산적인 것과 위상학적인 측면, 그리고 복잡한 기하 모델의 표현 및 처리들과 관련된 것들이 주를 이루고 있다^[1].

*정회원, 광주과학기술원
- 논문투고일: 2007. 01. 15
- 심사완료일: 2007. 07. 27

정확한 기하 모델을 만드는 과정에서 겪는 어려움들 중, 제일 중요한 것은 바로 위상학적으로 일관되고 수치적으로 정확한 기하 모델을 만드는 것이다. 여기서 '기하 모델이 위상학적으로 일관된다'라는 것은 만들어진 기하 모델이 크기나 모습에 관계없이 수학적으로 정의된 이상적인 모델과 같은 위상학적 구조를 갖는다는 의미이며, '수치적으로 정확한 기하 모델'은, 만들어진 모델이 이상적인 모델과 수치적으로 동등하다는 것을 의미한다.

보통 기하 모델은 여러 개의 평면 및 곡면의 조합으로 만들어지게 된다. 이의 구현을 위해서는 여러 형태를 갖는 곡면의 교차 곡선을 구해야 하는데, 특수한 경우를 제외하고는 일반적으로 정확한 교차 곡선을 구하는 것은 실제적으로 불가능하다¹⁾. 먼저 교차 곡선을 표현하는 다항식의 차수가 상당히 높아질 수 있다. 한 예로, 3차 텐서곱 곡면(bicubic tensor-product surface)들이 교차할 때, 그 정확한 교차선을 표현하는 곡선의 차수는 수학 곡선(algebraic curve)의 경우 324까지 올라갈 수 있으며, 매개변수 영역에서는 각 변수별로 차수가 54까지 이를 수 있다²⁾. 또한 부동소수점 연산시 컴퓨터에서 제공하는 제한된 정밀도로 인해서 발생하는 라운딩(rounding)에 의해서, 비록 정확한 교차 곡선 식을 구했다 하더라도, 실제 연산은 정확한 계산 값을 주지를 못하게 된다. 이의 해결을 위해서 exact arithmetic, symbolic arithmetic 등의 방법을 생각해 볼 수 있으나, 메모리 사용이 커지고 또 연산 시간이 많이 걸리기 때문에 실제 구현에는 사용되기 어렵다. 따라서 대부분의 경우 해결 방안으로 낮은 차수의 곡선을 이용해서 교차곡선을 근사시키는 방법이 많이 쓰이고 있다. 이 방법은 교차 곡선을 표현하는 다항식의 차수를 낮추기 때문에, 연산 시간 및 메모리 사용을 최소한으로 유지시킬 수 있다. 그렇지만, 이러한 방식은 기본적으로 정확한 표현이 아닌 근사 곡선을 이용하는 것이기 때문에, 교차 곡선 부위에서 필연적으로 오차를 발생시킨다. 근사 교차 곡선을 이용해서 모델을 생성할 경우, 결국 서로 이어지는 곡면 사이에 틈을 남기게 돼서, 제대로 된 기하 모델을 생성하지 못하게 된다³⁾.

이와 같은 틈이 존재하는 기하 모델은 특히 솔리드 모델을 정의할 때, 위상학적인 문제를 야기시킨다. 솔리드 모델은 수학적으로 모델의 내부, 경계 그리고 외부를 명확히 구분할 수 있어야 하는데, Fig. 1에서 보여진 것과 같은 틈이 존재할 경우, 내부와 외부의 구분이 모호해 지게 된다. 이는 곧, 위상학적으로 잘못된 모델을 의미하기 때문에, 이를 바탕으로 한 각종

계산이 의미가 없어지게 되고, 이후 신뢰할 수 없는 결과를 얻게 된다.

또한 이러한 결점이 있는 모델은 메시(mesh)를 생성할 때에도 문제가 된다. 현재 많이 쓰이고 있는 모든 메시 생성 프로그램들은 기본적으로 완벽한 기하 모델이 입력으로 주어지는 것을 기본으로 가정하고 있다. 그렇지만, 틈이 존재하는 결점이 있는 모델이 주어질 경우, 메시지를 생성하는 프로그램들은 보통 이러한 모델의 결점을 인지할 수 있는 기능이 없기 때문에, 잘못된 메시지를 생성할 수 있으며, 이를 바탕으로 한 해석 프로그램의 결과는 신뢰할 수 없고, 최악의 경우 해석 프로그램 자체가 멈추게 될 수 있다.

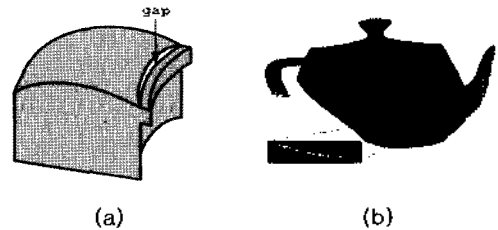


Fig. 1. (a) 연결되는 두 곡면간에 발생한 틈을 보여주는 그림. (b) 근사된 교차 곡선을 이용할 때 발생하는 틈을 보여주는 그림 ([5]에서 인용).

이러한 문제는 CFD(computational fluid dynamics)와 CAD프로그램간의 연결시 종종 발생한다⁴⁾. 이런 문제들 때문에 CAD 시스템이 보다 강력한 디자인과 개발 도구로써 사용되는 데에는 어려움이 있다. 이러한 모든 문제의 근원은 기하 모델링 단계에서 수치적으로나 위상학적으로 정확하지 못한 기하 모델의 생성에 기인하며, 이는 곧 현재의 모델링 시스템이 정확한 기하 모델을 생성할 경우 개발 시간의 단축 및 최적의 디자인을 자동으로 찾을 수 있다는 것을 의미한다. 이러한 문제들은 실제 산업 현장에서 대두되고 있지만, 그것들의 해결을 위한 연구는 전 세계적으로 미미한 실정이다. 특히 국내에서는 이와 관련된 인식과 기술이 부족한 실정이고 이에 따른 노력도 전무하다. 따라서 이러한 문제의 해결 및 관련 분야에서 선도적인 입지를 구축하기 위한 다각적인 이론 및 수치적인 연구가 필요하다.

2. 관련 연구 및 목적

기하 모델링과 메시 생성시 발생할 수 있는 오차 중 교차 곡선에 기인한 것이 큰 부분을 차지한다⁵⁾. 이는 수학적으로 정확한 교차곡선을 구하는 것은 실

제적으로 불가능하기 때문에 주로 발생하는 문제이다. 그렇지만 실제 응용의 관점에서 보면, 근사 곡선이 정확한 교차 곡선에 “충분히” 가까우면서 위상학적으로는 일관성을 갖추도록 하면 교차 곡선의 근사로 인한 오류 발생을 피할 수 있다^[1]. 그러므로 본 논문에서는 실제 응용의 범위 내에서 위상학적 측면과 수치적 정확성을 갖는 자유 곡면간의 근사 교차선과 기하 모델을 구현하는 방법의 개발과 그에 기반이 되는 이론적/수치적 기법에 대해서 논의한다. 먼저 연구 목표의 명확한 개념적인 이해를 돕기 위해 다음의 두 가지를 정의한다.

정의 1

위상학적 일관성 (topological consistency)이란 만들어진 모델(곡선 및 솔리드)이 이상적인 수학 모델과 위상학적으로 동일하다는 것으로 정의한다.

정의 2

수치적 정확성 (numerical accuracy)을 얻는다는 것은 주어진 오차 범위 내에서 위상학적 일관성을 유지하면서 수학적으로 정의된 정확한 기하 모델(곡선 및 솔리드)에 근사시키는 것으로 정의한다.

위상학적 일관성은 교차 곡선의 위상학적 일관성과, 교차 곡선 및 곡면으로 이루어지는 최종 기하 모델의 위상학적 일관성으로 구성되어 있다. 교차 곡선의 위상학적 일관성은 근사 곡선이 정확한 교차 곡선과 같은 위상학적 구조를 갖는지의 여부로 확인할 수 있다. 다음, 모델의 위상학적 일관성은 최종 모델이 수학적으로 정확히 정의된 모델과 위상학적으로 동일하지와 솔리드 모델의 경우에 있어서는 바로 최종 모델이 방수모델(watertight model)인지를 확인하는 것으로 알 수 있다. 여기서 방수 모델이라함은 모델에 어떠한 틈도 없어서 내부와 외부를 명확히 구분할 수 있는 모델을 의미한다. 이는 교차 부분에서 틈의 존재 여부로 판별할 수 있다. 따라서 주어진 오차범위 내에서 교차 곡선을 구하고 구한 근사 교차 곡선의 수치적 정확성을 이르면, 그것을 이용한 모델 또한 정확한 최종 모델과 위상학적으로 동일하게 된다.

지금까지 교차선에 대한 연구는 주로 수치적 정확성 측면과, 효율적인 방법의 개발에 집중되어 왔다^[6]. 특히 낮은 차수의 NURBS 곡선을 이용해서 교차선을 근사시키는 방법이 주목을 받아 왔다^[8]. 그렇지만, 국내외에서 교차선의 수치적 정확성과 위상학적 일관성을 동시에 고려한 문제가 관심을 끌게 된 것은 오래되

지 않았으며 관련 연구도 거의 이루어지지 않았다. Song 등^[9]은 처음으로 이러한 문제를 기술하였으며 이의 해결을 위한 방법을 제안하였다. 두 자유 곡면이 교차할 때, 교차 곡선을 중심으로 한 주변의 교차 곡면을 움직임으로써 교차 곡선에서 발생할 수 있는 틈을 오차 범위 내에서 제거한다. 주어진 조건에 따라서 C^1 또는 C^2 연속성을 갖도록 할 수 있다. 그렇지만, 일반적으로 재 설계 단계를 거치지 않는 한 최종 제품이 만들어 질 때까지 설계된 곡면에 인위적인 변형이 가해지는 것은 바람직하지 않다. 그렇지만, [5]에서 제안된 방법은 인위적으로 곡면을 변형 시켜야 하기 때문에, 예기치 못한 문제점을 야기시킬 수 있는 단점이 있다. Mukundan 등^[10]은 validated ODE방법을 곡면 교차곡선 문제에 적용하는 방법을 제시하였다. 그렇지만, 위상학적인 측면만을 고려한 교차 곡선의 오차 범위만 제시했을 뿐, 구체적인 교차곡선의 계산은 다루어지지 않았다.

본 논문에서는 교차 곡선 주변의 곡면을 변형시키지 않고 실제 응용의 범위 내에서 위상학적인 측면과 수치적인 측면을 동시에 만족시킬 수 있는 기하 모델을 구하기 위한 기반 문제 제기, 기반 이론 및 방법을 제안 하며, 특히 교차 곡선에 대해서 논의를 하고자 한다. 이를 위해 다음의 두 가지 세부 목표를 세우고:

- A. 위상학적 일관성을 위한 근사 교차 곡선의 오차 범위 결정.
 - B. 낮은 차수의 곡선을 이용한 교차 곡선의 근사 및 오차의 최소화.
- 이의 달성을 위해서 다음과 같은 방법을 제안한다.

3. 기술적 접근 방법

3.1 개요

위상학적으로 일관되고 수치적으로 정확한 교차 곡선을 근사시키기 위한 단계는 Fig. 2와 같이 구성할 수 있다.

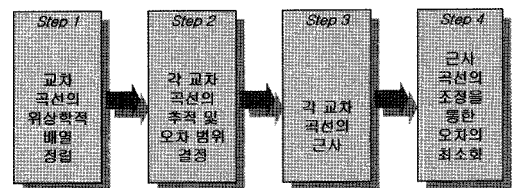


Fig. 2. 전체 흐름도.

3.2 교차 곡선의 위상학적 배열

두 곡면이 교차할 때 교차 곡선이 어떻게 구성 되

어 있는지는 미리 알지 못한다. 따라서, 교차 곡선이 폐곡선(loop)을 이루고 있는지, 하나의 곡선으로 구성 되어 있는지 아니면 두 갈래로 갈라지는 지점이 있는지 먼저 확인을 해야 한다. 이의 확인을 위해서는 위상학적인 구성을 살펴봐야 하며, 이는 곧 매개 변수 영역 내에서 교차 곡선의 위상학적 분포를 구함으로써 알 수 있다. 이를 위한 첫 번째 단계는 바로 임계점들(border, turning, singular)을 구하는 것이다. 이들 임계점들은 비선형 다항 방정식의 해의 형태로 나타나게 되며, 이를 구하기 위해서는 수치적으로 방정식의 해를 구하게 된다. 본 연구에서는 Interval Projected Polyhedron 방법을 이용한다^[7]. 각 임계점들을 구한 다음, [10,11]에 제시된 방법을 이용해서 교차곡선의 위상학적 분포를 구한다. 이렇게 위상학적인 배열을 구하면 복잡한 교차 곡선을 단순 증가 또는 단순 감소하는 영역으로 나누어 사용할 수 있게 된다.

3.3 교차 곡선의 추적 및 오차 범위 결정

본 단계에서는 앞서 구한 각 교차선의 시작과 끝점을 이용해서 각 교차선을 구하고 그에 상응하는 오차 범위들을 구한다. 각 교차선을 구하기 위해서는 미분 방정식을 이용한 추적법 (tracing method)을 이용한다^[7]. 교차 곡선을 나타내는 식은 비선형 미분방정식의 형태로 표현이 되며, 이를 풀기 위해서는 수치적 방법이 동원된다. 대표적인 방법으로, Runge-Kutta법이나, Adams-Bashforth법을 들 수 있다. 그렇지만, 이런 방법은 기본적으로 근사방법이며, 정확한 오차 범위를 주지 못할 뿐만 아니라 교차 곡선의 추적시 위상학적 문제를 야기시킬 수 있다^[9,10]. 특히 looping과 straying이 문제가 된다^[7,12]. 따라서 본 연구에서는 교차 곡선 계산시 위상학적 문제를 해결함과 동시에 정확한 해를 포함하는 오차 범위도 구할 수 있는 validated ODE 해법을 이용한다^[9]. 여기서 validated ODE는 validated Ordinary Differential Equation의 줄임 말로, 이 방법은 interval arithmetic을 근간으로 이루고 있으며, 매 계산 단계마다 해의 존재성/유일성이 보장되는 최소한의 영역을 삼아 가면서 해를 포함하는 영역을 찾아가는 수치 해석 방법이다. 일반적인 미분방정식 해법과 비교해서, 수치적으로나 위상학적으로 좋은 결과를 내주고 교차 곡선에서 발생하는 각종 수치적인 불안정성을 해결할 수 있는 장점이 있다^[9].

3.3.1 Validated ODE Solver

교차 곡선은 비선형 미분 방정식의 형태로 표현이

되며, 이를 풀기 위해서 validated ODE를 기반으로 한 풀이 방법을 이용한다^[9,13,14]. 이 방법은 정확한 해를 계산하지는 못하지만 해의 존재성과 유일성을 입증함과 동시에 정확한 해를 언제나 포함하는 오차 범위를 생성시킨다. 이 풀이법은 다음의 두 단계로 구성되어 있다:

Phase I

이 단계에서는 주어진 단계 (Fig. 3에서 (s_n, s_{n+1}) 또는 (s_{i+1}, s_{i+2}))에서 주어진 미분 방정식의 유일한 해가 존재하는 영역을 구한다. 이 영역은 Fig. 3에서 보여진 바와 같이 빗금 친 부분이며, 실제 정확한 미분 방정식의 해(비교를 위해서 Fig. 3에서 실선으로 표시됨. 실제로는 구해지지 않음)는 항상 이 영역 안에 존재하게 된다.

Phase II

이 단계에서는 Phase I에서 구한 매개 변수 s_i 에서의 정보를 바탕으로, s_{i+1} 에서 실제 정확한 미분방정식의 해가 가질 수 있는 값의 범위를 구하고 (Fig. 3에서 타원으로 표시), 또한 다음 단계에서 Phase I에서 구한 사각 영역의 크기를 최소화한다.

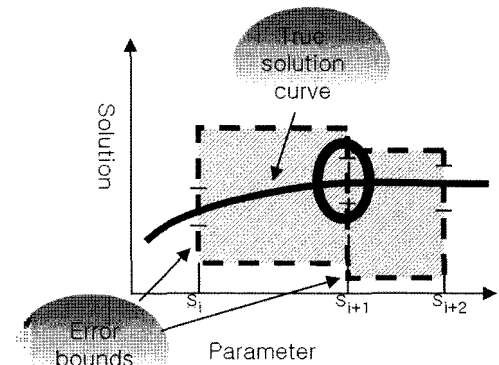


Fig. 3. Validated ODE기반 미분방정식 풀이법.

Phase I에서 구한 영역은 계산이 진행이 될수록 워핑(wrapping) 효과로 인해서 그 크기가 커지게 된다^[13]. 이를 최소로 줄이기 위해서 Phase II에서는 QR Factorization 방법을 이용하며^[13], 동시에 다음 단계에서 해의 영역을 구하기 위한 발판을 마련하게 된다.

Fig. 4는 validated ODE법을 기반으로 해서 미분 방정식의 해를 구하고 그 결과를 그래프로 나타낸 것이다. 그림에서 사각형들은 Phase I에서 구해진 해가 존재하는 영역을 나타내며, 이 영역은 오차 범위로 사용된다. 여기서 실선은 정확한 미분 방정식의 해이며

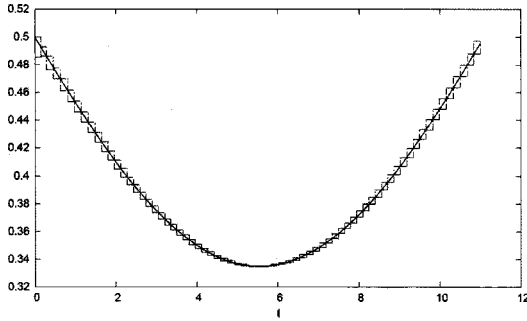


Fig. 4. Validated ODE 기반 미분방정식 해의 예. 점선은 오차 범위를 나타내며 실선은 수학적으로 구한 정확한 해를 나타낸다.

비교의 목적으로 표시되었다. 이 그림에서 보여진 예에서와 같이, 위의 두 단계를 반복적으로 적용하게 되면 미분 방정식의 해를 둘러싸는 영역을 구할 수 있으며, 이 영역 내에서 정확한 해가 유일 하게 존재함을 보장한다.

3.4 교차 곡선의 근사

본 단계에서는 앞서 구한 교차 곡선의 오차범위 내에서 정확한 교차 곡선을 근사시키는 낮은 차수의 곡선을 구한다. Fig. 5에서 보여진 바와 같이 교차 곡선은 매개변수 영역 (u, v) 또는 (σ, t) 에서 정의된 곡선들의 3차원 공간으로의 매핑(mapping)에 의해서 표현될 수 있다. Validated ODE기반의 추적법을 이용해서 교차 곡선을 구하면, 매개 변수 영역에서 정확한 교차 곡선 $K_1(s)$ 과 $K_2(s)$ 를 포함하는 오차 영역(사각형으로 표현)을 구하게 된다. 결국, 구해진 오차 범위 내에서 $K_1(s)$ 과 $K_2(s)$ 를 정확히 근사시키면, 매핑을 이용해서 3차원 공간 상에서의 근사 교차 곡선을 구할 수 있게 된다. 이의 근사를 위해서, 낮은 차수의 곡선을 사용하여 오차 범위 내에 위치하도록 한다. 이렇게 구해진 곡선은 그 오차 범위 영역 안에 포함되어 있기 때문에, 주어진 미분방정식의 근사해로 볼 수 있다. 수많은 곡선들 중에서 하나의 곡선을 구하기 위해서, 각 s_i 에서 범위로 주어진 매개 변수 값들의 중간 값을 선택하고, 이 중간 값들을 근사 시키는 곡선을 구한다. 본 논문에서는 B-spline을 이용한 근사곡선을 생각한다. 이 점들을 근사시키는 B-spline 곡선은 3차일 경우 노드벡터(knot vector) T 에 대해서

$$P(s) = \sum_{i=1}^m P_i N_{i,4}(s) \tag{1}$$

와 같이 표현이 되고, 그 근사 곡선의 계수들 P_i 들은

최소자승법에 의해서 구할 수 있다^[15,16]. 여기서 노드 벡터 T 는 주어진 점들의 분포를 고려해서 결정을 하게 되며, 이를 위해서 [17]에서 소개한 방법을 사용한다.

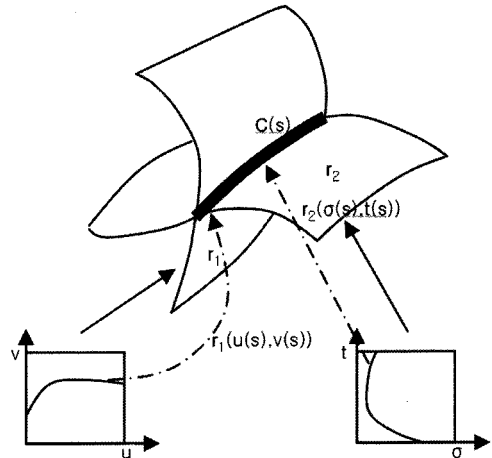


Fig. 5. 교차 곡선의 표현.

3.5 근사 곡선의 오차 최소화

일반적으로 두 곡면이 교차할 때, 세 가지의 서로 다른 교차 곡선을 표현하는 식들이 존재한다^[5]. Fig. 5에서 보인 바와 같이, 3차원 공간 상의 교차 곡선은 교차 곡선의 선호길이(arc length)를 매개변수로 이용한 $C_1(s)$ 와, 곡면 $r_1(u, v)$ 와 $r_2(\sigma, t)$ 상의 곡선의 형태로 얻어지는 $C_1(s) = r_1(u(s), v(s))$ 와 $C_2(s) = r_2(\sigma(s), t(s))$, 이렇게 세 가지의 형태로 각각 표현된다. 이상적인 경우, 이 세 곡선들은 하나의 곡선을 정의하게 된다. 그렇지만, 실제 연산에서는 컴퓨터 상의 부동소수점연산에 기인한 제약과 근사 곡선의 사용으로 인해서 세 표현간에 오차가 발생하는 것을 피할 수 없다. 이러한 오차는 근사 곡선을 정확한 교차 곡선과 ‘충분히’ 가깝게 근사 시킴으로써 최소화하는 것이 가능하며, 이와 같은 근사 곡선을 바탕으로 실제 응용 범위 내에서 모델의 위상학적 일관성을 유지시키고, 또한 수치적

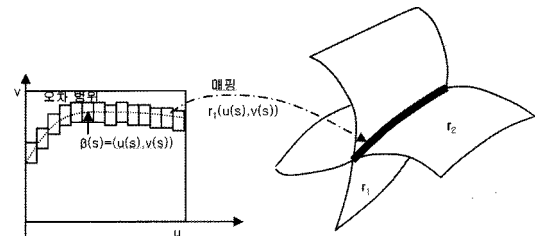


Fig. 6. 매개 변수 영역 내 교차 곡선과 오차 범위.

정확성도 동시에 얻는 기하 모델을 구할 수 있게 된다. 이러한 근사로 인한 오차를 줄이기 위해서, 먼저 3.4절에서 제시된 방법을 이용해서 근사 곡선 구한 후, 이것의 오차를 줄이는 방법을 사용한다.

3.5.1 최적화를 통한 교차 곡선의 조절

각 매개 변수 영역에서 구해진 근사 곡선들을 오차 범위 내에서 움직여 가면서, $C_1(s)$ 와 $C_2(s)$ 간의 차이를 줄일 수 있으며, 이렇게 함으로써 정확한 교차 곡선에 가까이 근사하는 곡선을 구할 수 있다.

두 개의 교차하는 B-spline곡면 $r_1(u,v)$ (차수 m, n) 와 $r_2(\sigma, t)$ (차수 l, p)가 있다고 가정한다. 여기서 u, v, σ, t 는 0과 1사이이고 s 는 교차 곡선의 선호길이를 나타내는 매개변수로 가정한다. 이러한 가정하에서 각 매개변수 영역 ($u-v$ 와 $\sigma-t$) 안에서 교차 곡선을 나타내는 식은 각각 다음과 같이 주어진다.

$$K_1(s) = (u(s), v(s)) \tag{2}$$

$$K_2(s) = (\sigma(s), t(s)) \tag{3}$$

이들 이용하면 3차원 공간상에서 교차곡선은 다음과 같이 표현되며

$$C_1(s) = r_1(u(s), v(s)) \tag{4}$$

$$C_2(s) = r_2(\sigma(s), t(s)) \tag{5}$$

여기에 관련이 된 오차함수 $E(s)$ 는 다음과 같다.

$$E(s) = |C_1(s) - C_2(s)| \tag{6}$$

이상적인 조건하에서는 두 교차 곡선식은 동일한 하나의 곡선을 표현해야 하기 때문에 식 (6)은 영(0)이 되어야 한다. 그렇지만 보통의 경우 이를 만족시키는 것은 불가능하기 때문에 대신 그 값을 최소로 만드는 K_1 과 K_2 를 찾는 것을 목표로 한다. 식 (6)을 다음과 같이 쓸 수 있으며

$$f = (C_1(s) - C_2(s)) \cdot (C_1(s) - C_2(s)) \tag{7}$$

결국 이 함수를 최소로 만드는 K_1 와 K_2 를 구하는 것으로 문제를 정의한다.

3.5.2 해결법

함수 (7)을 최소화 하는 문제를 풀기 위해서 두 가

지의 방법을 고려할 수 있다. 첫 번째는 먼저 K_1 와 K_2 를 바탕으로 합성함수(function composition)방법을 이용해서 푸는 방법이다. 그렇지만 함수를 합성하기 때문에 식이 상당히 복잡해지고, 따라서 계산 시간도 많이 소요된다. 두 번째 방법은 위의 최적화 문제를 근사최적화 문제로 변환해서 푸는 방법이며 이 방법을 본 논문에서는 사용하였다.

앞서 소개한 validated ODE법을 이용하면, 각 s_i 에 대한 u_i, v_i, σ_i, t_i 값을 구하게 된다. 이를 이용해서 다음과 같은 목적함수를 구할 수 있다.

$$f^* = \sum_{i=1}^w (C_1(s_i) - C_2(s_i)) \cdot (C_1(s_i) - C_2(s_i)) \tag{8}$$

여기서 w 는 s_i 점들의 개수가 된다. 식 (8)은 식 (7)과 수학적으로는 동등하지 않다. 그렇지만, 점의 개수 w 가 증가하면 식 (7)에 근접하게 된다. 이 최적화 문제는 제한 조건하에서 고려되어야 한다. 3.3.1절에서 소개된 validated ODE방법을 사용하게 되면, 정확한 해를 포함하는 영역을 알 수 있다. 따라서 식 (8)을 최소화 하기 위하여 K_1 과 K_2 를 정의하는 독립변수들을 조절해야 하는데, 이것을 통해서 계산되는 u, v, σ, t 는 구해진 오차범위 이내에 반드시 포함되어야 한다. 이를 최적화에 제한 조건으로 적용하기 위해서, 각 s_i 에서 u_i, v_i, σ_i, t_i 각각에 해당하는 범위를 이용한다. 이 범위는 validated ODE방법 중, 3.3.1절의 Phase II에서 구할 수 있다. 따라서 이를 식으로 표현하면 다음과 같다.

$$\begin{aligned} u_i^l \leq u_i \leq u_i^h, v_i^l \leq v_i \leq v_i^h \\ \sigma_i^l \leq \sigma_i \leq \sigma_i^h, t_i^l \leq t_i \leq t_i^h \end{aligned} \tag{9}$$

여기서 윗 첨자 l 과 h 는 각각 낮은 경계값(lower bound)과 높은 경계값(upper bound)를 의미한다. 이와 같은 제한조건 아래에서 목적함수 (8)을 최소화 하는 K_1 과 K_2 의 세수(B-spline곡선의 경우 제어점(control point))를 구하기 위해서 본 논문에서는 L-BFGS-B 최적화 방법을 이용한다^[18]. 이 최적화 기법은 독립변수의 수가 많을 때 좋은 성능을 보여준다고 알려져 있으며^[18] 본 계산에 적합하다.

3.5.3 조절된 곡선의 검증

다음 단계는 조절된 근사 곡선이 오차 범위 내에 포함되는지 검증하는 것이다. 본 검증을 위해서는 다음의 과정을 거친다. 먼저 구해진 각 곡선 K_1 과 K_2 를 각 점 s_i 에서 노트(knot)삽입법을 이용해서 작은 단위

로 쪼갬다. 이렇게 나누어진 각 곡선의 제어점이 계산된 오차범위 내에 있는지를 확인한다. 만일 오차 범위 내에 없으면, 구해진 곡선이 오차범위 내에 들어있지 않다고 결정한 다음, 제어점의 수를 늘린 새로운 K_1 과 K_2 를 계산해서 앞서 소개한 과정을 다시 반복한다. 보통의 경우 (9)의 구속조건을 만족하는 근사곡선은 오차범위 내에 포함된다. 이렇게 검증이 된 교차 곡선은 정확한 교차곡선과 위상학적으로 동등하게 된다.

4. 예 제

본 장에서는 앞서 소개한 알고리즘을 이용해서 실제 교차 곡선을 구하는 예제들을 소개한다. 이 계산을 위해서 500MHz CPU와 128Mb RAM을 가진 Linux 시스템을 이용하였다.

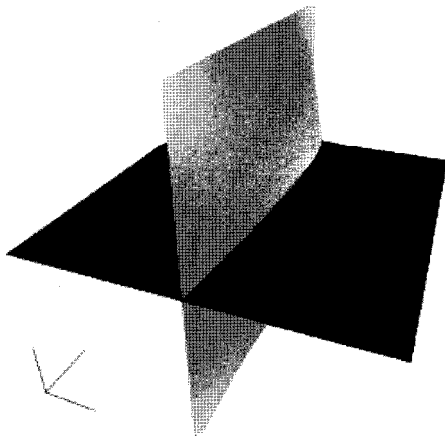


Fig. 7. 교차하는 두 베지에 곡면.

첫 번째 예제로 Fig. 7에 보여준 교차하는 두 베지에 곡면 $P(\sigma, t)$ 와 $Q(u, v)$ 를 이용 하였다. 교차 곡선은 하나의 곡선으로 구성되어 있다. 시작점은 Interval Projected Polyhedron algorithm^[7]을 이용해서 구하면 네 개의 interval $(\sigma, t, u, v) \dots ([0.0, 0.0], [0.49999, 0.50001], [0.0, 0.0], [0.49999, 0.50001])$ 로 나타난다. Validated ODE solver를 이용해서 교차 곡선을 구하면, s_i 마다 각 매개 변수 값들을 구할 수 있다. 그 중 Fig. 8은 매개 변수 t 에 대한 결과를 나타낸다. $u = 0.5, \sigma = 0.5$ 지점에서 임계점이 발생하며, 이 지점에서 교차 곡선을 나누어서 단순 변화하는 부분에 해당하는 교차 곡선을 포함하는 오차범위를 구한다. 구해진 s_i 의 개수는 311개이며, $u-v$ 영역에서 이를 표시하면 Fig. 9로 나타난다.

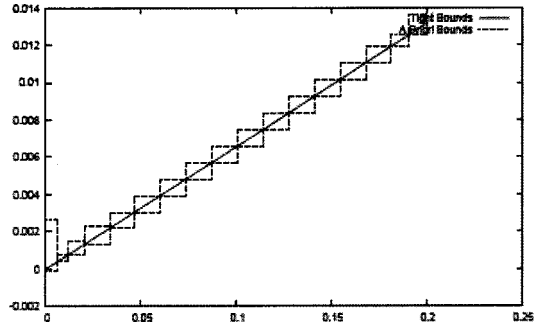


Fig. 8. 매개변수 t에 대한 오차영역.

먼저 s_i 에서 각 매개변수 값들을 이용해서 각각 제어점 10개를 이용한 3차 B-spline곡선을 기반으로 근사 K_1 과 K_2 를 최소 자승법을 이용해서 결정한다. s_i 에서 매개 변수값은 구속조건 (9)에서 주어지는 범위의 중간 값을 택한다. 이 근사된 K_1 과 K_2 를 이용하고 3차원 공간상에서 식 (6)을 이용해서 교차곡선의 오차 값을 구한다.

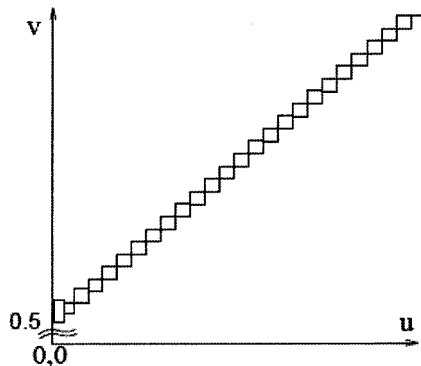


Fig. 9. $u-v$ 영역에서 오차범위(전체 그림의 일부만 표시).

오차값은 각 s_i 범위인 $[s_i, s_{i+1}]$ 안에서 적어도 세 개 이상의 점들을 샘플하여 계산하였다. 본 예제의 경우 전부 1241개의 샘플된 점에서 식 (6)을 계산하였다. 구해진 오차의 최대값은 5.571×10^{-5} , 평균값은 4.776×10^{-6} 이었다. 최적화를 한 다음에 같은 점에서 계산된 오차의 최대값은 6.32×10^{-5} 이었고, 평균값은 6.85×10^{-6} 이었다. 최적화를 하면서 최대 오차값은 소폭 증가를 하였지만, 평균 오차값은 대폭 감소가 되었다. 두 번째 실험은 30개의 제어점을 이용해서 K_1 과 K_2 를 구해서 최대 오차의 변화를 구해보았다. 최적화 전의 최대 오차값은 4.78×10^{-5} 이었고, 평균은 4.77×10^{-6} 이었다. 최적화를 한 다음 최대 오차값은 $6.31 \times$

10^{-5} 이었으며 평균 오차값은 1.88×10^{-6} 으로 줄어드는 결과를 얻었다. 본 실험에서 알 수 있는 점은 제어점의 개수를 고정시킨 상태에서 그 위치를 조절함으로써 전반적인 오차를 줄일 수 있다는 것이다. 즉 제어점의 수가 늘어나면 그만큼 곡선을 조절 할 수 있는 자유도가 증가하게 되므로 좀 더 세밀하게 근사 곡선을 교차 곡선에 근접시킬 수 있다는 것을 관찰 할 수 있다. 두 실험에서 비록 최대 오차값은 소폭 증가를 하였지만, 그 지점에 노트 삽입(knot insertion)을 통해서 제어점의 수를 증가 시킴으로써 최대 오차값을 줄일 수 있다.

다음 예제는 단순하게 변하는 교차곡선을 선택하지 않았을 경우 어떤 현상이 일어나는지를 보여주는 예제이다. 이를 위해서 Fig. 10에서 보여진 바와 같이 교차하는 3차/2차 곡면들을 이용하였다. Fig. 11과 Fig. 12는 validated ODE법을 이용해서 구한 $u-v$ 와 $\sigma-t$ 영역에서의 교차곡선이다. 본 그림에서는 s_t 에서 각 매개변수 영역의 중간값을 식선으로 연결해서 그래프로 표시 하였다. K_1 과 K_2 를 위한 3차 B-spline곡선의 제어점으로 각각 46개와 39개를 사용하였고, 최적화하기 전의 교차 곡선과의 최대와 평균 오차값은 각각 0.0004533과 0.0001732이었다. 최적화를 한 이후의 오차의 최대 값과 평균값은 각각 0.0005208과 0.0001402가 되었다. 이 경우에는 최적화를 하여도 오차값이 줄어드는 정도가 미미하였는데, 그 이유는 회전인계점(turning point) ($u=0.5, \sigma=0.5$)에서 곡선이 급격히 변하기 때문에, 주어진 제어점(control point)의 수로는 정밀하게 표현이 안되기 때문이다.

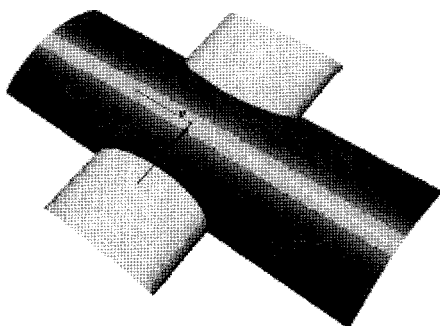


Fig. 11. 교차하는 3차/2차 베지에 교차 곡면.

사실 이러한 경우는 처리과정에서 임계점으로 걸러지게 되며 두 개의 단순 변화 곡선들로 나뉘게 된다. 즉, 실제 교차곡선을 근사시킬 때에는 $u-v$ 와 $\sigma-t$ 영역에서 단순 변화하는 부분만을 다루게 되므로 본 예제

에서 관찰된 문제를 피할 수 있으며, 전체 오차를 줄일 수 있다.

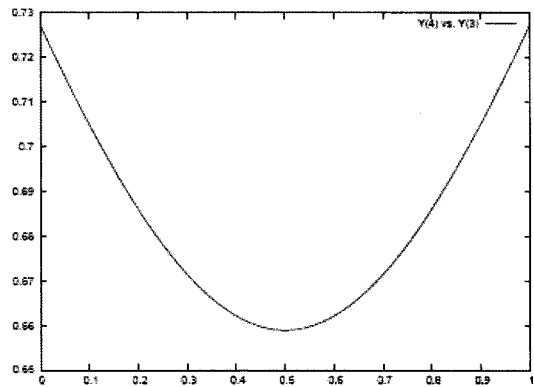


Fig. 12. 교차곡선을 나타내는 $u-v$ 곡선.

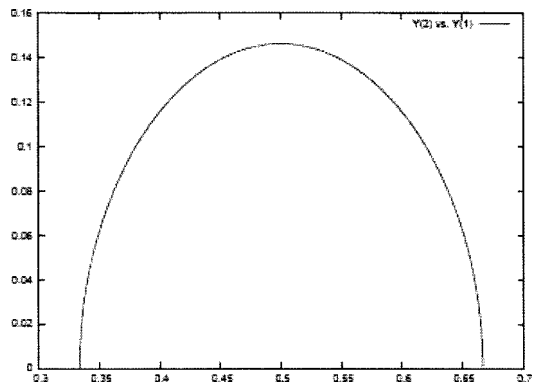


Fig. 13. 교차곡선을 나타내는 $\sigma-t$ 곡선.

5. 결 론

본 연구에서는 기하 모델링에서 위상학적인 측면과 정밀도 측면을 동시에 고려한 교차 곡선에 대해서 기존 방법의 문제점을 논하였고, 그것을 해결하기 위한 이론적/실제적 방법을 제안하였으며, 제안된 방법의 가능성에 대해서 논하였다. 교차 곡선의 위상학적인 일관성을 위해서 validated ODE 기반의 추적기법을 이용해서 교차 곡선의 오차범위를 구하였고, 낮은 차수의 곡선을 이용해서 교차 곡선을 근사시킨 다음, 정확한 교차 곡선과의 오차를 줄이기 위해서 최적화 기법을 이용해서 근사 곡선을 조절하였다. 이와 같은 방법을 통해서 정확한 교차 곡선과 위상학적으로 같은 근사 곡선을 선택할 수 있고, 최적화 기법을 통해서 근사된 교차 곡선의 정밀도를 향상시킬 수 있게 되며, 기하 모델링의 강인성(robustness)을 향상시킬 수

있는 가능성을 보였다.

복잡한 곡면이 주어졌을 경우 그것이 만들어 내는 교차 곡선도 복잡한 형태를 보여주게 된다. 이럴때, 본 논문에서 제시된 방법은 다음과 같이 적용될 수 있다. 두 곡면이 교차할 경우, 제일 먼저 해야 할 일은 3.2절에 제시된 방법을 이용해서 교차 곡선의 위상학적인 배열을 구하는 것이다. 이 과정을 통해서, 각 교차 곡선 중, 단변화(단순 증가/단순 감소) 하는 교차 곡선의 성분을 구할 수 있게 된다. 이렇게 각각 구해진 교차 곡선의 성분에 본 논문에서 제시된 방법을 적용하면 최대 근접 근사 교차 곡선을 계산할 수 있으며, 결국 전체 교차 곡선을 구할 수 있게 된다.

제안된 방법에는 몇 가지 고려해야 할 사항이 있다. 먼저, 각 교차 곡선들간, 그리고 교차되는 곡면들간의 연속성에 대한 부분이다. 각 교차 곡선 중, 단순 변화하는 부분들을 독립적으로 택해서 근사 교차곡선을 구하기 때문에, 서로 인접한 교차곡선들의 C^0 연속성만 고려되었다. 또한, 교차 하는 곡면들 사이에서도 C^0 연속성만 고려되었다. 기하 모델을 제작할 때 상황에 따라서 높은 차수의 연속성도 고려가 되어야 하기 때문에 $C^n (n \geq 0)$ 연속성을 이루기 위한 공식이 필요하다. 또 다른 문제점으로, 3차 B-spline을 이용한 교차 곡선의 근사이다. 일반적으로 교차 곡선은 대수학(algebraic) 또는 비대수학(non-algebraic) 곡선일 수 있기 때문에 3차 B-spline으로 정확하게 근사시키는 것은 수학적으로 어렵다. 또한 매개화에서 고려되어야 할 사항은 주어진 매개 변수(u, v, σ, t)들에 대해서 선형 길이로 주어진 매개 변수 값이 고정되어 있기 때문에, 보다 정확한 근사가 어려운 면이 있다는 점이다. 따라서 보다 더 정밀한 근사 교차 곡선을 구하기 위해서 위에 열거한 문제점들을 해결해야 하는 것이 중요하며, 이를 위한 후속 연구가 계획되어 있다.

감사의 글

이 논문은 2006년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2006-331-D00036).

참고문헌

1. Farouki, R. T., "Closing the Gap between CAD Model and Downstream Application", *SIAM News*,

- Vol. 32, No. 5, pp. 1-3, 1999.
2. Sederberg, T. W., Implicit and Parametric Curves and Surfaces for Computer Aided Geometric Design, PhD Thesis, Purdue University, 1983.
 3. Katz, S. and Sederberg, T. W., "Genus of the Intersection Curve of Two Rational Surface Patches", *Computer Aided Geometric Design*, Vol. 5, pp. 253-258, 1988.
 4. Shen, G., Analysis of Boundary Representation Model Rectification, PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA, 2000.
 5. Song, X., Sederberg, T. W., Zheng, J., Farouki, R. T. and Hlass, J., "Linear Perturbation Methods for Topologically Consistent Representations of Free-form Surface Intersections", *Computer Aided Geometric Design*, Vol. 21, No. 3, pp. 303-319, 2004.
 6. Hoschek, J. and Lasser, D., Fundamentals of Computer Aided Geometric Design, A K Peters, Wellesley, MA, 1993.
 7. Patrikalakis, N. M. and Maekawa, T., Shape Interrogation for Computer Aided Design and Manufacturing, Springer, New York, 2002.
 8. Bajaj, C. L. and Xu, G., "NURBS Approximation of Surface/Surface Intersection Curves", *Advances in Computational Mathematics*, Vol. 2, pp. 1-21, 1994.
 9. Mukundan, H., Ko, K. H., Maekawa, T., Sakkalis, and Patrikalakis, N. M., "Tracing Surface Intersections with a Validated ODE System Solver", Proceedings of the 9th EG/ACM Symposium on Solid Modeling and Applications, G. Elber, N. Patrikalakis and P. Brunet, editors. pp. 249-254. Genoa, Italy, Eurographics Press, 2004.
 10. Sakkalis, T., "The Topological Configuration of a Real Algebraic Curve", *Bulletin of the Australian Mathematical Society*, Vol. 43, pp. 37-50, 1991.
 11. Grandine, T. A. and Klein, F. W., "A New Approach to the Surface Intersection Problem", *Computer Aided Geometric Design*, Vol. 14, pp. 111-134, 1997.
 12. Patrikalakis, N. M., Maekawa, T., Ko, K. H. and Mukundan, H., "Surface to Surface Intersections", *Computer-Aided Design & Applications*, Vol. 1, No. 1-4, pp. 449-458, 2004.
 13. Lohner, R. J., "Computation of Guaranteed Enclosures for the Solutions of Ordinary Initial and Boundary Value Problem", In Computational Ordinary Differential Equations, Cash J., Gladwell I., (Eds.), Clarendon Press, Oxford, pp. 425-235, 1992.
 14. Nedialkov, N. S., Jackson, K. R. and Corliss, G. E., "Validated Solutions of Initial Value Problems for Ordinary Differential Equations", *Applied Mathematics and Computation*, Vol. 105, No. 1, pp. 21-68, 1999.

15. Plegl, L. A. and Tiller, W., The Nurbs Book, Springer, New York, 1995.

16. Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P., Numerical Recipes in C, Cambridge University Press, Cambridge, 1988.

17. Park, H., "An Error-bounded Approximate Method for Representing Planar Curves in B-splines". *Computer Aided Geometric Design*, Vol. 21, pp. 479-497, 2004.

18. Zhu, C., Byrd, R. H., Lu, P. and Nocedal, J., L-BFGS-B: FORTRAN Subroutines for Large Scale Bound Constrained Optimization. Tech. Report NAM-11, EECS Department, Northwestern University, 1994.



고 광 회

1995년 서울대학교 조선해양공학과 학사
 1995년~1997년 삼성중공업
 1998년~2001년 M.I.T. 기계공학과-해양공학과 석사
 2001년~2003년 M.I.T. 해양공학과 박사
 2003년~2004년 M.I.T. Seagrant College Program Post Doc.

2004년~2005년 Design and Manufacturing Institute, Stevens Institute of Tech. 연구원
 2006년~현재 광주과학기술원 기전공학과 조교수
 관심분야: Geometric Modeling, CAD/CAM, Physics Based Modeling, Computer Graphics, DDDAS